

Query Answering in Ontologies under Preference Rankings

İsmail İlkan Ceylan

Theoretical Computer Science
Technische Universität Dresden, Germany

Rafael Peñaloza

KRDB Research Centre
Free University of Bozen-Bolzano, Italy

Thomas Lukasiewicz

Department of Computer Science
University of Oxford, UK

Oana Tifrea-Marcuska

The Alan Turing Institute
London, UK

Abstract

We present an ontological framework, based on preference rankings, that allows users to express their preferences between the knowledge explicitly available in the ontology. Using this formalism, the answers for a given query to an ontology can be ranked by preference, allowing users to retrieve the most preferred answers only. We provide a host of complexity results for the main computational tasks in this framework, for the general case, and for \mathcal{EL} and $DL-Lite_{core}$ as underlying ontology languages.

1 Introduction

Description logics (DLs) [Baader *et al.*, 2007] are a family of knowledge representation formalisms that have been successfully used for modeling many real-world domains. Important recent applications include semantic search on the Web and ontology-based access to data, including Big Data [Giese *et al.*, 2015]. One crucial reasoning task for the above areas and other knowledge domains is conjunctive query (CQ) answering, which corresponds to computing all tuples of individuals that satisfy some conceptual pattern. One of the issues related to CQ answering over ontologies is to be able to manage the large number of potential answers in a structured manner. Since in standard ontological CQ answering, all answers are qualitatively indistinguishable, it is, e.g., impossible to filter the most preferred answers to a given CQ. It is thus important to extend CQ answering over ontologies with preference criteria, such as preference rankings.

Example 1. A preference ranking may encode Bob’s preferences over sources of information on the Web: Bob is not a fan of blogs; so, he prefers any non-blog source of information over blogs. But when reading blogs, Bob wants more subjective opinions, so, blogs written by a non-specialist are preferred over blogs written by a specialist. If Bob reads information from a non-blog source written by a non-specialist, then he prefers popular sources over non-popular ones, otherwise, non-popular sources over popular ones. ■

In this paper, preference modeling is done via very general preference rankings over a collection of possible choices, with the only restriction (for our computational complexity results) that the rank of each choice is computable in

polynomial time. Indeed, many rankings in information retrieval (IR) have this property (see, e.g., [Joachims, 2002]). Thus, the approach in this paper actually provides very general results for combining DLs with IR rankings.

We consider preferences that are directly associated with the axioms and facts in a knowledge base. Such “absolute” rankings are actually quite common in practice, in particular, they are broadly used in Web applications; e.g., Google’s PageRank is also directly associated with Web pages. We implement this idea by annotating every piece of knowledge with a context, which intuitively describes the situations in which this knowledge holds, and by defining a unique preference ranking over these contexts. These preferences are then naturally extended to answers to CQs, allowing users to retrieve only the most preferred answers.

Example 2. An ontological knowledge base extracted from information sources on the Web can be coupled with Bob’s preferences in Example 1 by annotating ontological axioms with events, over which we define a suitable preference ranking. Thus, a travel ontology can be associated with contexts, e.g., to express that popular blogs ($b \wedge p$) recommend that an itinerary with a wine destination would work well with another wine destination, and that specialist blogs ($b \wedge s$) say that Sicily is a wine destination, each with a suitable rank. Then, the answers “Florence”, “Sicily”, and “Bordeaux” for an ontological query asking for wine and spa destination may then be ranked as 1, 0.35, and 0, respectively, depending on the underlying preference ranking. ■

Annotating the knowledge with contexts has been previously used successfully in, e.g., probabilistic logic programming [Poole, 1997] and probabilistic databases [Suciu *et al.*, 2011]. The main benefit of using contexts, rather than providing a preference ranking directly to the knowledge is that they provide an easily accessible interface to the knowledge. Thus, if different users want to express different preferences, e.g., related to different query circumstances, then they only need to provide a new preference ranking over the same contexts, without modifying the underlying knowledge base.

This paper’s main contributions are briefly as follows:

- We propose ranked ontologies as a novel approach to modeling the preferences of a user relative to the knowledge in an ontology. They are based on unique preference rankings, which extend to query answers, so that only the most preferred answers are given, ordered via their ranks.

– We then provide generic complexity results for deciding k most preferred answers to a CQ for different types of complexities. We also provide complexity results for this problem for the lightweight DLs \mathcal{EL} and $DL-Lite_{core}$, which include especially also tractability and first-order rewritability results.

– We also give generic complexity results for other important reasoning problems, namely, for deciding k most preferred conditional answers, for deciding a lower bound for the preference degree of a Boolean CQ (BCQ), and for deciding k most preferred worlds. Moreover, we give complexity results for these problems for \mathcal{EL} and $DL-Lite_{core}$, which include further tractability and first-order rewritability results.

The rest of this paper is organized as follows. Section 2 defines preference rankings and recalls the basic concepts of description logics. In Section 3, we introduce ranked ontologies. Section 4 provides alternative characterizations for ranked ontologies and the ranks of BCQs. In Section 5, we define the main reasoning tasks for these ontologies, while Section 6 provides tight complexity results for these tasks. We conclude with a discussion on related work, a summary of the main results, and an outlook on future work. Due to space limitations, detailed proofs of all results in this paper will be given in an extended paper.

2 Preliminaries

In this section, we define preference rankings and briefly recall description logics (DLs) [Baader *et al.*, 2007].

Preference Rankings. Given a nonempty finite set V of Boolean variables, a *valuation* \mathcal{W} of V assigns to each $X \in V$ a truth value among *true* and *false*, also abbreviated as X and $\neg X$, respectively. A *valuation* \mathcal{U} for a set of variables $U \subseteq V$ assigns to each $X \in U$ a truth value among *true* and *false*. We denote by Ω the set of all valuations of V . A *preference ranking* over Ω is a function $\text{rank}: \Omega \rightarrow [0, 1]$, which is extended to any $\Gamma \subseteq \Omega$ by $\text{rank}(\Gamma) = \sup_{\omega \in \Gamma} \text{rank}(\omega)$, where “sup” is the supremum (i.e., the least upper bound). Intuitively, $\text{rank}(\Gamma)$ represents the degree of preference in Γ . Here, we only consider rankings on valuations where each rank is computable in polynomial time, which is a property that is shared by many rankings in IR (see, e.g., [Joachims, 2002]).

Such a polynomial preference ranking is also naturally defined via possibilistic networks (PNs), which compactly encode possibility distributions via (possibilistic) independencies encoded in a directed acyclic graph (DAG) [Benferhat *et al.*, 1999] in a similar way as Bayesian networks [Pearl, 1988; Darwiche, 2009] compactly encode probability distributions. Note that despite these similarities, PNs are semantically and computationally very different from Bayesian networks (see also Section 7). Note also that PNs are only one example of encoding polynomial preference rankings.

Example 3. A *possibilistic network* (PN) $\mathcal{P} = (G, \Phi)$ over V consists of a DAG $G = (V, E)$ and a set Φ containing a conditional possibility distribution $\text{rank}_{\mathcal{P}}(x | \text{pa}(x))$ for every $x \in V$ given $\text{pa}(x)$, where $\text{pa}(x)$ denotes the parents of x in G (i.e., the immediate predecessors of x in G). Note that each conditional possibility distribution $\text{rank}_{\mathcal{P}}(x | \text{pa}(x))$ consists of one conditional possibility value $\text{rank}_{\mathcal{P}}(x | \text{pa}(x))$

for each pair of valuations \mathbf{x} and $\text{pa}(\mathbf{x})$ of x and $\text{pa}(x)$, respectively. Such a PN defines a unique (joint) possibility distribution over the valuations \mathcal{W} of V (where \mathbf{x} and $\text{pa}(\mathbf{x})$ are matching valuation of x and $\text{pa}(x)$, respectively): $\text{rank}_{\mathcal{P}}(\mathcal{W}) = \prod_{x \in V} \text{rank}_{\mathcal{P}}(\mathbf{x} | \text{pa}(x))$. For example, Figure 1 shows a possibilistic network \mathcal{P}_0 over the variables $V_0 = \{b, s, p\}$. The tables associated with each node contain the conditional possibility distributions for this node given its parents. For example, the node b is associated with an unconditional possibility distribution, since it has no parents, while p is associated with a distribution conditional on b and s . Here, the possibility of, e.g., the valuation $\{b, s, \neg p\}$ (i.e., $b = s = \text{true}$ and $p = \text{false}$) is $\text{rank}_{\mathcal{P}_0}(\{b\}) \cdot \text{rank}_{\mathcal{P}_0}(\{s\} | \{b\}) \cdot \text{rank}_{\mathcal{P}_0}(\{\neg p\} | \{b, s\}) = 0.7 \cdot 0.5 \cdot 1 = 0.35$. ■

Note also that PNs can be used to compactly encode the conditional preferences of a user (i.e., statements of the form “if x holds, then $\neg y$ is preferred over y ”) over a finite set of events [Ben Amor *et al.*, 2014]. Intuitively, for each conditional event, the user provides a possibility degree (i.e., a rank) that is proportional to the user’s preference of its occurrence. The joint possibility distribution then combines the ranks of all conditional events to a ranking over the valuations of the variables in V . The following example shows that the above possibilistic network from Figure 1 in fact represents the conditional preferences described in Example 1. For further (and larger) examples of how finite sets of conditional preferences can be encoded as possibilistic networks, see, e.g., [Ben Amor *et al.*, 2014; Amor *et al.*, 2015].

Example 4. The PN \mathcal{P}_0 from Figure 1 expresses the preferences of Bob over sources of information on the Web when planning his trips. Bob is not a fan of blogs, therefore, he (unconditionally) prefers any other source of information than blogs ($\neg b$) over blogs (b). Bob wants more subjective opinions when reading blogs (b), therefore, he prefers a blog written by a non-specialist ($\neg s$) over a blog written by a specialist (s). Note that these are examples of conditional preferences, where the order between s and $\neg s$ depends on the choice made for the evaluation of the variable b before. If Bob reads information not from a blog written by a non-specialist ($\neg b \neg s$), then he prefers a popular source (p) over a non-popular source ($\neg p$), otherwise, the non-popular source ($\neg p$) is preferred over popular ones (p). Overall, e.g., $\{b, s, \neg p\}$ is preferred over $\{b, s, p\}$ (since $0.7 \cdot 0.5 \cdot 1 > 0.7 \cdot 0.5 \cdot 0.5$). ■

Description Logics. We briefly sketch some basics in description logics (DLs) [Baader *et al.*, 2007]. In DLs, the knowledge of an application domain is represented through an *ontology* \mathcal{O} , which is a finite set of *axioms* that restrict the possible interpretations that can be given to the terms used. Ontologies are usually partitioned into a set of *terminological axioms* (called *TBox*) that encode the relations between the different terms used in the knowledge domain, and a set of *assertional axioms* (called *ABox*) that express the knowledge about specific individuals. The semantics of DLs is given via *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a nonempty set, called *domain*, and $\cdot^{\mathcal{I}}$ is the interpretation function that describes how the terms of the ontology are interpreted. A *satisfaction relation* “ \models ” defines which interpretations \mathcal{I} satisfy which axioms A , denoted $\mathcal{I} \models A$. We say that \mathcal{I} *satisfies* (or

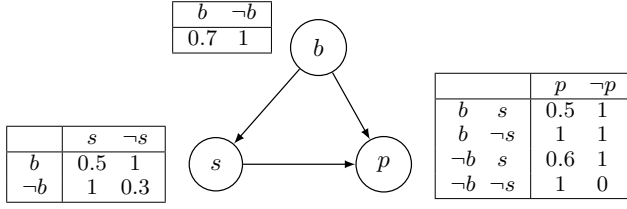


Figure 1: Preference ranking \mathcal{P}_0 encoded via a possibilistic network (PN) over $V_0 = \{b, s, p\}$.

is a *model* of) an ontology \mathcal{O} , if \mathcal{I} satisfies all axioms in \mathcal{O} .

An important reasoning task for DLs is conjunctive query answering. A *conjunctive query* (CQ) $q(\mathbf{x})$ is of the form $\exists \mathbf{y} \bigwedge_{i=1}^n p_i(\mathbf{x}, \mathbf{y})$, where the $p_i(\mathbf{x}, \mathbf{y})$'s are atoms over individual names and two disjoint sets of variables \mathbf{x} and \mathbf{y} . A *Boolean CQ* (BCQ) q is a CQ of the form $q()$. An *answer* θ for a CQ $q(\mathbf{x})$ to an ontology \mathcal{O} maps each variable in \mathbf{x} to an individual name such that $q(\mathbf{x}\theta)$ holds in all models of \mathcal{O} . The *answer* for a BCQ q to \mathcal{O} is *true* (resp., *false*), denoted $\mathcal{O} \models q$, if $\theta = \emptyset$ is an (resp., not an) answer for q to \mathcal{O} .

3 Ranked Ontologies

In this section, we introduce ranked ontologies, which are a novel combination of preference rankings with ontologies. We consider an arbitrary description logic (DL) [Baader *et al.*, 2007] as underlying ontology language.

Syntax. A ranked ontology associates every axiom in an ontology with a context, encoded by a propositional formula over a set of variables V , which intuitively describes the situation in which the axiom is guaranteed to hold. Additionally, a preference ranking over these contexts is given.

In the sequel, let V be a finite nonempty set of Boolean variables, and \mathcal{L} be a DL. A *V-context* φ is a propositional formula over V . A *V-axiom* $\langle \lambda : \varphi \rangle$ in \mathcal{L} consists of an axiom λ in \mathcal{L} and a *V-context* φ . A *V-ontology* in \mathcal{L} is a finite set of *V-axioms* in \mathcal{L} . A *ranked ontology* $\mathcal{K} = (\mathcal{P}, \mathcal{O})$ in \mathcal{L} over V consists of a preference ranking \mathcal{P} over V and a *V-ontology* \mathcal{O} in \mathcal{L} . We often omit the prefix V . As in the classical case, *V-ontologies* are partitioned into a (*V*-)TBox and a (*V*-)ABox. Note that classical ontologies are a special case of *V-ontologies*, where all *V-axioms* are of the form $\langle \lambda : \top \rangle$ with \top denoting *true*. The *restriction* of a ranked ontology $\mathcal{K} = (\mathcal{P}, \mathcal{O})$ to a valuation \mathcal{W} of the variables in V is the classical ontology $\mathcal{O}_{\mathcal{W}} = \{\lambda \mid \langle \lambda : \varphi \rangle \in \mathcal{O}, \mathcal{W} \models \varphi\}$ in \mathcal{L} .

Example 5. A ranked ontology $\mathcal{K}_0 = (\mathcal{P}_0, \mathcal{O}_0)$ is given by the preference ranking \mathcal{P}_0 encoded in the PN of Figure 1 and \mathcal{O}_0 below. Intuitively, it says, e.g., that popular blogs ($b \wedge p$) recommend that an itinerary with a wine destination would work well with another wine destination, and that specialist blogs ($b \wedge s$) say that Sicily is a wine destination:

$$\begin{aligned} \mathcal{O}_0 = \{ & \langle \exists \text{worksWellWith.WineDest} \sqsubseteq \text{WineDest} : b \wedge p \rangle, \\ & \langle \text{WineDest} \sqsubseteq \text{RelaxDest} : \neg b \vee s \rangle, \\ & \langle \exists \text{worksWellWith.RelaxDest} \sqsubseteq \text{SpaDest} : \neg s \rangle, \\ & \langle \text{RelaxDest} \sqsubseteq \text{SpaDest} : \neg p \rangle \} \cup \\ & \{ \langle \text{SpaDest}(\text{florencia}) : s \rangle, \langle \text{WineDest}(\text{sicily}) : b \wedge s \rangle, \\ & \langle \text{WineDest}(\text{bordeaux}) : b \wedge \neg s \rangle \}. \quad \blacksquare \end{aligned}$$

Semantics. We next extend the classical interpretations of \mathcal{L} to additionally evaluate contexts. A *contextual interpretation* is a pair $(\mathcal{I}, \mathcal{W})$, where $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is a classical interpretation for \mathcal{L} , and \mathcal{W} is a valuation of V . We say that $(\mathcal{I}, \mathcal{W})$ *satisfies* (or is a *model* of) the axiom $\langle \lambda : \varphi \rangle$, denoted $(\mathcal{I}, \mathcal{W}) \models \langle \lambda : \varphi \rangle$, if either (i) $\mathcal{W} \not\models \varphi$, or (ii) $\mathcal{I} \models \lambda$. Notice that “(i) or (ii)” is equivalent to “ $\mathcal{W} \models \varphi$ implies $\mathcal{I} \models \lambda$ ”: intuitively, λ is only required to hold within the context φ . We say that $(\mathcal{I}, \mathcal{W})$ *satisfies* (or is a *model* of) a *V-ontology* \mathcal{O} over \mathcal{L} , denoted $(\mathcal{I}, \mathcal{W}) \models \mathcal{O}$, if it satisfies all axioms in \mathcal{O} .

We now define ranked interpretations over contextual ones, and the satisfaction of ranked ontologies $\mathcal{K} = (\mathcal{P}, \mathcal{O})$ in them. Intuitively, the contexts connect the preference ranking \mathcal{P} to the *V-ontology* \mathcal{O} and thus define a set of ranked interpretations represented by \mathcal{K} . Formally, a *ranked interpretation* $\mathfrak{I} = (\mathfrak{J}, \text{rank})$ consists of a finite set of contextual interpretations \mathfrak{J} and a ranking rank over \mathfrak{J} (which assigns a rank $\text{rank}((\mathcal{I}, \mathcal{W}))$ to each $(\mathcal{I}, \mathcal{W}) \in \mathfrak{J}$). We say that \mathfrak{I} is a *model* of a *V-ontology* \mathcal{O} , if every $(\mathcal{I}, \mathcal{W}) \in \mathfrak{J}$ satisfies \mathcal{O} ; it is a *model* of \mathcal{P} , if for each valuation \mathcal{W} , $\max_{(\mathcal{I}, \mathcal{W}) \in \mathfrak{J}} \text{rank}((\mathcal{I}, \mathcal{W})) = \text{rank}_{\mathcal{P}}(\mathcal{W})$. We say \mathfrak{I} is a *model* of $\mathcal{K} = (\mathcal{P}, \mathcal{O})$, denoted $\mathfrak{I} \models \mathcal{K}$, if it is a model of \mathcal{O} and \mathcal{P} . We say \mathcal{K} is *consistent*, if it has at least one model.

Example 6. Consider again the ranked ontology $\mathcal{K}_0 = (\mathcal{P}_0, \mathcal{O}_0)$ of Example 5. Let $\mathcal{I}_0 = (\{d, e, f\}, \cdot^{\mathcal{I}_0})$ be a DL interpretation with $\text{florencia}^{\mathcal{I}_0} = d$, $\text{sicily}^{\mathcal{I}_0} = e$, $\text{bordeaux}^{\mathcal{I}_0} = f$, $\text{WineDest}^{\mathcal{I}_0} = \text{RelaxDest}^{\mathcal{I}_0} = \{d, e, f\}$, $\text{SpaDest}^{\mathcal{I}_0} = \emptyset$, and $\text{worksWellWith}^{\mathcal{I}_0} = \emptyset$. Then, the contextual interpretation $(\mathcal{I}_0, \{-b, s, p\})$ is a model of \mathcal{O}_0 , since $\text{florencia}^{\mathcal{I}_0} \in \text{WineDest}^{\mathcal{I}_0}$ and $\text{WineDest}^{\mathcal{I}_0} \subseteq \text{RelaxDest}^{\mathcal{I}_0}$, while $(\mathcal{I}_0, \{-b, s, -p\})$ is not a model of \mathcal{O}_0 , as it does not satisfy $\langle \text{RelaxDest} \sqsubseteq \text{SpaDest} : \neg p \rangle$. A ranked interpretation $\mathfrak{I} = (\mathfrak{J}, \text{rank})$ that satisfies the KB \mathcal{K}_0 is then given by the singleton set $\mathfrak{J} = \{(\mathcal{I}_0, \{-b, s, p\})\}$ with $\text{rank}((\mathcal{I}_0, \{-b, s, p\})) = 0.6$. \blacksquare

We next define the rank of BCQs q under ranked ontologies $\mathcal{K} = (\mathcal{P}, \mathcal{O})$, which is intuitively the most strict rank of q under all ranked interpretations $\mathfrak{I} = (\mathfrak{J}, \text{rank})$ that satisfy \mathcal{K} . Formally, the *rank* of q under $\mathfrak{I} = (\mathfrak{J}, \text{rank})$, denoted $\text{rank}_{\mathfrak{I}}(q)$, is defined by $\text{rank}_{\mathfrak{I}}(q) = \max_{(\mathcal{I}, \mathcal{W}) \in \mathfrak{J}, \mathcal{I} \models q} \text{rank}((\mathcal{I}, \mathcal{W}))$, while the *rank* of q under \mathcal{K} is then defined by: $\text{rank}_{\mathcal{K}}(q) = \inf_{\mathfrak{I} \models \mathcal{K}} \text{rank}_{\mathfrak{I}}(q)$.

In general, we are not only interested in the rank of a given BCQ, but also in its rank given some partial knowledge of the current context. Conversely, given a BCQ, we are also interested in the most preferred source that entails it. For these two tasks, we extend ranks to contexts. The *rank* of q and a context φ under $\mathfrak{I} = (\mathfrak{J}, \text{rank})$, denoted $\text{rank}_{\mathfrak{I}}(q \wedge \varphi)$, is defined as follows: $\text{rank}_{\mathfrak{I}}(q \wedge \varphi) = \max_{(\mathcal{I}, \mathcal{W}) \in \mathfrak{J}, \mathcal{I} \models q, \mathcal{W} \models \varphi} \text{rank}((\mathcal{I}, \mathcal{W}))$, while the *rank* of q and φ under \mathcal{K} , denoted $\text{rank}_{\mathcal{K}}(q \wedge \varphi)$, is defined by $\text{rank}_{\mathcal{K}}(q \wedge \varphi) = \inf_{\mathfrak{I} \models \mathcal{K}} \text{rank}_{\mathfrak{I}}(q \wedge \varphi)$. We define the *conditional rank* of a query given a context, and of a context given a query, using the standard product conditioning rule, as follows:

$$\begin{aligned} \text{rank}_{\mathcal{K}}(q \mid \varphi) &= \text{rank}_{\mathcal{K}}(q \wedge \varphi) / \text{rank}_{\mathcal{K}}(\varphi), \text{ if } \text{rank}_{\mathcal{K}}(\varphi) > 0, \\ \text{rank}_{\mathcal{K}}(\varphi \mid q) &= \text{rank}_{\mathcal{K}}(q \wedge \varphi) / \text{rank}_{\mathcal{K}}(q), \text{ if } \text{rank}_{\mathcal{K}}(q) > 0. \end{aligned}$$

4 Semantic Results

We now provide alternative semantic characterizations for the consistency of ranked ontologies and for the rank of BCQs and contexts in consistent ranked ontologies.

Consistency. The following informally shows that a ranked ontology \mathcal{K} over V is consistent iff, for every valuation \mathcal{W} of V with positive rank, the restriction of \mathcal{K} to \mathcal{W} is consistent.

Theorem 1. *A ranked ontology $\mathcal{K} = (\mathcal{P}, \mathcal{O})$ in \mathcal{L} over V is consistent iff, for every valuation \mathcal{W} of V , if $\text{rank}_{\mathcal{P}}(\mathcal{W}) > 0$, then $\mathcal{O}_{\mathcal{W}}$ is consistent.*

Thus, the consistency of ranked ontologies in \mathcal{L} can be reduced to the consistency of classical ontologies in \mathcal{L} . As every ontology in \mathcal{EL} is consistent [Baader *et al.*, 2005], the theorem implies that every ranked ontology in \mathcal{EL} is consistent.

Example 7. Consider the ranked ontology $\mathcal{K}_1 = (\mathcal{P}_0, \mathcal{O}_1)$ built in *DL-Lite_{core}*, where \mathcal{P}_0 is defined by the PN from Figure 1, and $\mathcal{O}_1 = \{\langle \text{WineDest} \sqsubseteq \neg \text{WineDest} : \neg b \wedge \neg p \rangle, \langle \text{WineDest}(\text{florence}) : s \rangle\}$. For $\mathcal{W}_1 = \{\neg b, s, \neg p\}$, it holds that $\text{rank}_{\mathcal{P}_0}(\mathcal{W}_1) = 1 > 0$ (see Figure 1 above). Moreover, the ontology $\mathcal{O}_{\mathcal{W}_1} = \{\text{WineDest} \sqsubseteq \neg \text{WineDest}, \text{WineDest}(\text{florence})\}$ is inconsistent. Hence, \mathcal{K}_1 is also inconsistent. However, for $\mathcal{O}_2 = \{\langle \text{WineDest} \sqsubseteq \neg \text{WineDest} : \neg b \wedge \neg p \rangle, \langle \text{WineDest}(\text{florence}) : \neg s \rangle\}$, the ranked ontology $\mathcal{K}_2 = (\mathcal{P}_0, \mathcal{O}_2)$ is consistent, although for $\mathcal{W}_2 = \{\neg b, \neg s, \neg p\}$, the restriction $\mathcal{O}_{\mathcal{W}_2}$ is inconsistent. Indeed, for the *DL-Lite* interpretation $\mathcal{I}_2 = (\{d\}, \mathcal{I}_2)$, where $\text{WineDest}^{\mathcal{I}_2} = \emptyset$ and $\text{florence}^{\mathcal{I}_2} = d$, the ranked interpretation $\mathfrak{P} = (\mathfrak{J}, \text{rank})$ given by $\mathfrak{J} = \{(\mathcal{I}_2, \mathcal{W}) \mid \mathcal{W} \neq \mathcal{W}_2\}$, and for all $\mathcal{W} \neq \mathcal{W}_2$, $\text{rank}((\mathcal{I}_2, \mathcal{W})) = \text{rank}_{\mathcal{P}_0}(\mathcal{W})$ satisfies \mathcal{K}_2 . ■

Rank. Informally, the next theorem shows that, for consistent ranked ontologies \mathcal{K} , the rank of a BCQ q can be determined by looking at the restrictions of \mathcal{K} that entail q , and that this result can also be extended to contexts.

Theorem 2. *For every consistent ranked ontology $\mathcal{K} = (\mathcal{P}, \mathcal{O})$ in \mathcal{L} over V , BCQ q , and context φ over V :*

$$\begin{aligned} \text{rank}_{\mathcal{K}}(q) &= \max_{\mathcal{O}_{\mathcal{W}} \models q} \text{rank}_{\mathcal{P}}(\mathcal{W}), \text{ and} \\ \text{rank}_{\mathcal{K}}(q \wedge \varphi) &= \max_{\mathcal{W} \models \varphi, \mathcal{O}_{\mathcal{W}} \models q} \text{rank}_{\mathcal{P}}(\mathcal{W}). \end{aligned}$$

Note that the precondition that the ranked ontology \mathcal{K} is consistent is fundamental for this theorem to hold. Consider, e.g., the inconsistent ranked ontology \mathcal{K}_1 from Example 7, and let $q_1 = \{A(b)\}$. Then, by definition, since \mathcal{K}_1 has no models, $\text{rank}_{\mathcal{K}_1}(q_1) = 1$ (since it is the infimum of an empty subset of $[0, 1]$). However, $\max_{\mathcal{O}_{\mathcal{W}} \models q_1} \text{rank}_{\mathcal{P}_0}(\mathcal{W}) = 0$.

A direct consequence of Theorem 2 is that there are finitely many (at most $2^{|V|} + 2$) possible ranks: the rank of a BCQ q corresponds to $\text{rank}_{\mathcal{P}}(\mathcal{W})$ for some valuation \mathcal{W} of V , or 0 if q is not entailed by any restriction $\mathcal{O}_{\mathcal{W}}$, or 1, if the ranked ontology is inconsistent.

5 Reasoning Tasks

In this section, we formally define the main reasoning tasks for ranked ontologies, namely deciding whether the rank of a BCQ is above a threshold (called p -entailment), top- k CQ answering, top- k conditional CQ answering, and computing the k most preferred worlds for a BCQ.

Table 1: Most preferred answers (resp., worlds) for the CQ $q_2(\chi)$ (resp., BCQ $q_2(\chi\theta_0)$) to \mathcal{K}_0 from Example 8.

Worlds	θ_0	θ_1	θ_2	$\text{rank}_{\mathcal{P}_0}(\mathcal{W})$	$\text{rank}_{\mathcal{P}_0}(\mathcal{W} \mid q_2(\chi\theta_0))$
$\mathcal{W}_0 = \{b, s, p\}$	×	×	×	0.175	-
$\mathcal{W}_1 = \{b, s, \neg p\}$	✓	✓	×	0.35	0.35
$\mathcal{W}_2 = \{b, \neg s, p\}$	×	×	×	0.7	-
$\mathcal{W}_3 = \{b, \neg s, \neg p\}$	×	×	×	0.7	-
$\mathcal{W}_4 = \{\neg b, s, p\}$	×	×	×	0.6	-
$\mathcal{W}_5 = \{\neg b, s, \neg p\}$	✓	×	×	1	1
$\mathcal{W}_6 = \{\neg b, \neg s, p\}$	×	×	×	0.3	-
$\mathcal{W}_7 = \{\neg b, \neg s, \neg p\}$	×	×	×	0	-

p -Entailment. The p -entailment problem is informally the problem of deciding whether the entailed rank of a BCQ under a ranked ontology is above a given threshold. Formally, given a ranked ontology \mathcal{K} , a BCQ q , a context φ , and some $p \in (0, 1]$, decide whether $\text{rank}_{\mathcal{K}}(q \wedge \varphi) \geq p$ holds.

Top- k Answers. As for more general CQs $q(\mathbf{x})$ to ranked ontologies $\mathcal{K} = (\mathcal{P}, \mathcal{O})$, since \mathcal{P} represents preferences, we are especially interested in most preferred answers, which are the ones with highest ranks. A *top- k answer*, where $k \in \mathbb{N}$ is fixed, for $q(\mathbf{x})$ to \mathcal{K} is a tuple $(\theta_1, \dots, \theta_k)$ of different answers θ_i for $q(\mathbf{x})$ to \mathcal{K} such that either (a) $\theta_1, \dots, \theta_l$ with $l \leq k$ are the only answers for $q(\mathbf{x})$ to \mathcal{K} , or (b) the following conditions (i) and (ii) hold:

- (i) for all $i, 1 \leq i < k$: $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_i)) \geq \text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_{i+1}))$;
- (ii) for no other answer θ : $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_k)) < \text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta))$.

As different answers may have the same rank, top- k answers are not unique, i.e., there may be different tuples satisfying the properties of a top- k answer, and they may also be empty.

Top- k Conditional Answers. In some cases, we have some information about the context in which we are currently. Thus, it is also important to find the answers that are most preferred, given a context φ . A *top- k answer*, where $k \in \mathbb{N}$ is fixed, for $q(\mathbf{x})$ under a context φ to \mathcal{K} is a tuple $(\theta_1, \dots, \theta_l)$ of $l \in \{0, \dots, k\}$ different answers θ_i for $q(\mathbf{x})$ to \mathcal{K} such that either (a) $\theta_1, \dots, \theta_l$ with $l \leq k$ are the only answers for $q(\mathbf{x})$ to \mathcal{K} , or (b) the following conditions (i) and (ii) hold:

- (i) for all $i, 1 \leq i < k$: $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_i) \mid \varphi) \geq \text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_{i+1}) \mid \varphi)$;
- (ii) for no other answer θ : $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_k) \mid \varphi) < \text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta) \mid \varphi)$.

As $\text{rank}_{\mathcal{K}}(\varphi)$ is a constant factor for all $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_i) \mid \varphi)$, it is sufficient to compare the ranks of all $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta_i) \wedge \varphi)$.

k Most Preferred Worlds. Similarly to finding k most preferred answers to a CQ, we may also be interested in finding k most preferred valuations for a BCQ. Intuitively, they are the best k sources for a BCQ, i.e., they are the k most preferred arguments for justifying a BCQ. Given a BCQ q , a ranked ontology \mathcal{K} over V , a fixed $k \in \mathbb{N}$, and different propositional valuations $\mathcal{W}_i, 1 \leq i \leq k$ over V , we say that $\mathcal{W}_1, \dots, \mathcal{W}_k$ are *k most preferred worlds* for q to \mathcal{K} , if (i) and (ii) hold:

- (i) for all $i, 1 \leq i < k$: $\text{rank}_{\mathcal{K}}(\mathcal{W}_i \mid q) \geq \text{rank}_{\mathcal{K}}(\mathcal{W}_{i+1} \mid q)$;
- (ii) for no other valuation \mathcal{W} : $\text{rank}_{\mathcal{K}}(\mathcal{W}_k \mid q) < \text{rank}_{\mathcal{K}}(\mathcal{W} \mid q)$.

Example 8. Consider again the ranked ontology $\mathcal{K}_0 = (\mathcal{P}_0, \mathcal{O}_0)$ of Example 5 and the CQ $q_2(\chi) = \text{WineDest}(\chi) \wedge \text{SpaDest}(\chi)$. Then, three possible answers for q_2 to \mathcal{K}_0 exist: $\theta_0 = \{\chi/\text{florence}\}$, $\theta_1 = \{\chi/\text{sicily}\}$, and $\theta_2 = \{\chi/\text{bordeaux}\}$,

Table 2: Complexity of the ranked extension of \mathcal{EL} (all entries without “in” are completeness results).

Problem	data	KB	ranking	combined
p -entailment	in P	in P	NP	NP
top- k (conditional) answers	in P	in P	Δ_2^p	Δ_2^p
k most preferred worlds	in P	in P	CONP	CONP

Table 3: Complexity of the ranked extension of $DL-Lite_{core}$ (all entries without “in” are completeness results).

Problem	data	KB	ranking	combined
p -entailment	in AC^0	NLOGSPACE	NP	NP
top- k (conditional) answers	in AC^0	in P	Δ_2^p	Δ_2^p
k most preferred worlds	in AC^0	in coNLOGSPACE	CONP	CONP

having the ranks

$$\begin{aligned} \text{rank}_{\mathcal{P}_0}(q_2(\chi\theta_0)) &= \max\{\text{rank}_{\mathcal{P}_0}(\mathcal{W}_1), \text{rank}_{\mathcal{P}_0}(\mathcal{W}_5)\} = 1, \\ \text{rank}_{\mathcal{P}_0}(q_2(\chi\theta_1)) &= \text{rank}_{\mathcal{P}_0}(\mathcal{W}_1) = 0.35, \text{ and} \\ \text{rank}_{\mathcal{P}_0}(q_2(\chi\theta_2)) &= \text{rank}_{\mathcal{P}_0}(\emptyset) = 0, \end{aligned}$$

respectively; see Table 1. Note that θ_2 has the rank 0, as \mathcal{K}_0 does not entail $q_2(\chi\theta_2)$ in any valuation. It is easy to see from the table that (θ_0, θ_1) is a top-2 answer, and that \mathcal{W}_5 and \mathcal{W}_1 are the two most preferred worlds for $q_2(\chi\theta_0)$. ■

6 Complexity Results

In this section, we provide complexity results for ranked ontologies, both generic ones and for the special cases of the lightweight DLs \mathcal{EL} [Baader *et al.*, 2005] and $DL-Lite_{core}$ [Calvanese *et al.*, 2007; Artale *et al.*, 2009].

As for the complexity of query answering in DLs, one usually considers some parts of the input as fixed. In particular, one speaks of the *data* (resp., *KB*) *complexity* when only the size of the ABox (resp., the whole ontology) is considered, and of the *combined complexity* when also the query is part of the input. Note that, in our framework, the ontology is enriched with a ranking, but the two are rather decoupled by the use of contexts. In fact, the size of the ranking does not usually grow proportionally in the size of the data/KB, i.e., the same annotations can be coupled with many axioms of the ontology. Therefore, the ranking is fixed in both the data and the KB complexity. We also consider the *ranking complexity* when only the size of the preference ranking is relevant. In the sequel, the *combined complexity* refers to the whole input, including the preference ranking. For an arbitrary but fixed DL \mathcal{L} , we denote by \mathcal{C}_d , \mathcal{C}_k , and \mathcal{C}_c , the data, KB, and combined complexity of BCQ answering in \mathcal{L} , respectively.

We recall that in \mathcal{EL} , BCQ entailment is polynomial in the data and KB complexity, but NP-complete in the combined complexity [Rosati, 2007]; in $DL-Lite_{core}$, these complexities are membership in AC^0 , NLOGSPACE-completeness, and NP-completeness, respectively [Artale *et al.*, 2009].

The results for the ranked extensions of \mathcal{EL} and $DL-Lite_{core}$ are summarized in Tables 2 and 3, respectively: reasoning in these DLs is in general tractable in the data and KB complexity, but intractable in the ranking and combined complexity.

p -Entailment. The following theorem provides complexity results for p -entailment; its proof is based on Theorem 2.

Theorem 3. *The p -entailment problem for ranked ontologies in \mathcal{L} is in \mathcal{C}_d , \mathcal{C}_k , and NP^{C_c} in the data, KB, and combined complexity, respectively, and is NP-complete in the ranking complexity. Furthermore, if \mathcal{C}_c is contained in NP, then it is NP-complete in the combined complexity.*

In particular, for ranked ontologies in \mathcal{EL} , p -entailment is polynomial in the data and KB complexity, and NP-complete in the ranking and combined complexity. For $DL-Lite_{core}$, this problem is in AC^0 , NLOGSPACE-complete, and NP-complete in the respective complexities.

Note that these complexity results do not necessarily hold for *conditional* p -entailments, i.e., deciding whether $\text{rank}_{\mathcal{K}}(q|\varphi) \geq p$, which is in fact harder in the network complexity. Assuming that $\text{rank}_{\mathcal{P}}(\varphi) > 0$, we have $\text{rank}_{\mathcal{K}}(q|\varphi) = \text{rank}_{\mathcal{K}}(q \wedge \varphi) / \text{rank}_{\mathcal{P}}(\varphi)$. Thus, before we can decide whether $\text{rank}_{\mathcal{K}}(q|\varphi) \geq p$, it is necessary to compute $\text{rank}_{\mathcal{P}}(\varphi)$, whose associated decision problem is already hard for the second level of the polynomial hierarchy. This is stated in the following theorem, which follows from a reduction from the problem of finding the maximum satisfying assignment of a set of weighted clauses [Krentel, 1988].

Theorem 4. *Given a preference ranking \mathcal{P} , $p \in [0, 1]$, and a propositional formula φ , deciding whether $\text{rank}_{\mathcal{P}}(\varphi) = p$ is Δ_2^p -hard.*

Top- k Answers. If the size of the CQ $q(\mathbf{x})$ and the preference ranking \mathcal{P} are fixed, then there are polynomially many possible answers for $q(\mathbf{x})$ to $\mathcal{K} = (\mathcal{P}, \mathcal{O})$. For each such answer θ , by Theorem 2, we can compute $\text{rank}_{\mathcal{K}}(q(\mathbf{x}\theta))$ by performing constantly many BCQ entailment tests in \mathcal{L} . If \mathcal{P} is in the input, the problem becomes Δ_2^p -hard, even for simple instance queries, by Theorem 4. But it remains in Δ_2^p in the combined complexity if classical BCQ entailment is in the first level of the polynomial hierarchy.

Theorem 5. *Let $A = (\theta_1, \dots, \theta_k)$ be a tuple of answers for a CQ $q(\mathbf{x})$ to a ranked ontology \mathcal{K} . If \mathcal{C}_d contains P, then deciding whether A is a top- k answer is in \mathcal{C}_d , \mathcal{C}_k , and $(\Delta_2^p)^{C_c}$ in the data, KB, and combined complexity, respectively, and Δ_2^p -complete in the ranking complexity. If \mathcal{C}_c is contained in NP, then it is Δ_2^p -complete in the combined complexity.*

In particular, for ranked ontologies in \mathcal{EL} , top- k query answering is in P in the data and KB complexity, and Δ_2^p -complete in the ranking and combined complexity. Note that the complexity results in Table 3 for ranked ontologies in $DL-Lite_{core}$ are obtained via separate proofs. In particular, in the data complexity, one can build a first-order query verifying that \mathfrak{A} is already a top- k answer, which is in AC^0 .

Most Preferred Worlds. As we are interested in valuations, rather than arbitrary contexts, computing conditional ranks is easier. Indeed, $\text{rank}_{\mathcal{P}}(\mathcal{W})$ is computable in polynomial time. Hence, if BCQ answering in \mathcal{L} is polynomial, one can decide in polynomial time whether $\text{rank}_{\mathcal{K}}(\mathcal{W}|q) \geq \text{rank}_{\mathcal{K}}(\mathcal{W}'|q)$ for any two valuations \mathcal{W} and \mathcal{W}' . Hardness follows, if \mathcal{P} is part of the input, from the fact that deciding the existence of some \mathcal{W} with $\text{rank}_{\mathcal{P}}(\mathcal{W}) \geq p$ is already NP-hard.

Theorem 6. *Deciding whether $\mathcal{W}_1, \dots, \mathcal{W}_k$ are k most preferred worlds for the BCQ q is in C_d, C_k , and coNP^{C_c} in the data, KB, and combined complexity, resp., and coNP -complete in the ranking complexity. If C_c is contained in NP, then it is coNP -complete in the combined complexity.*

In particular, for ranked ontologies in \mathcal{EL} , deciding k most preferred worlds is in P in the data and KB complexity, and coNP -complete in the ranking and combined complexity. For $DL\text{-Lite}_{\text{core}}$, this problem is in AC^0 , in NLOGSPACE, and coNP -complete in the respective complexities.

7 Related Work

A different combination of DLs with preferences for ranking objects is presented in [Lukasiewicz and Schellhase, 2007], where conditional preferences define a ranking function that allows to perform a semantic personalized search and ranking over a set of resources annotated via an ontological description. In [Lukasiewicz *et al.*, 2013], Datalog+/- is extended with preferences closely related to those previously studied for relational databases. A similar combination of DLs with purely qualitative preferences is the approach in [Di Noia *et al.*, 2013], which combines DLs with CP-nets in such a way that variable values of CP-nets are satisfiable DL formulas, and that ontological axioms are used to restrict CP-net outcomes. The work [Di Noia *et al.*, 2015], like ours, also deals with computing k most preferred answers to CQs, but differently from ours, it is again based on CP-nets and existential rules. Although CP-nets are also graphical models for describing preferences, they differ greatly from polynomial preference rankings (and even possibilistic networks), both in their expressivity and in their computational complexity (in CP-nets, deciding dominance is PSPACE-complete, rather than polynomial). Another interesting approach to mixing qualitative preferences with Semantic Web technology is [Siberski *et al.*, 2006], where SPARQL is extended to encode user preferences in the query.

Generalizing possibilistic logic [Dubois and Prade, 2004], Hollunder [1995], Dubois *et al.* [2006], and Liao and Yao [2001] define possibilistic extensions of DLs, with applications in information retrieval. Similarly, a model for information retrieval based on possibilistic directed networks is proposed in [Brini *et al.*, 2005]. Possibilistic extensions of DLs are also used for handling inconsistencies in ontologies [Qi *et al.*, 2011]. All these approaches generalize standard first-order interpretations to possibilistic ones and interpret pairs of ontological axioms and possibilistic weights in them. Here, instead, we connect DLs under standard first-order interpretations via contexts to unique preference rankings, which may be encoded as possibilistic networks. Borgwardt *et al.* [2016] use possibilistic networks to define a ranking on all answers to an ontological query, rather than an “absolute” ranking on the knowledge base; their framework and complexity results are based on existential rules, rather than on \mathcal{EL} and $DL\text{-Lite}_{\text{core}}$. In [HadjAli *et al.*, 2011; Dubois *et al.*, 2013], preferences are handled via possibilistic logic, while our work is on preference-based ontological query answering, combining ontologies and preference rankings (potentially encoded as possibilistic networks).

Less closely related, probabilistic DLs [d’Amato *et al.*, 2008; Ceylan and Peñaloza, 2017; Ceylan and Peñaloza, 2015] may similarly be context-based combinations of DLs with unique probability distributions, such as those in Bayesian networks. In [Lukasiewicz *et al.*, 2014], probabilistic preference logic networks allow for dealing with preferences under probabilistic uncertainty in Markov random fields. However, polynomial preference rankings are very different from probability distributions. In particular, the rank of an event is the *maximum* of the ranks of all satisfying worlds, while its probability is the *sum* of their probabilities. Therefore, inference with preference rankings is computationally much easier than in Bayesian networks or other probabilistic graphical models (see also [Borgelt and Kruse, 2003]).

8 Summary and Outlook

We have introduced ranked ontologies as a general framework for extending DLs with a unique preference ranking, where each rank is computable in polynomial time, as a method for representing and reasoning about users’ conditional preferences about ontological knowledge. Using this approach, users may retrieve only the most preferred answers to a given query, instead of being overwhelmed by a large number of potentially irrelevant answers. We have provided a host of complexity results for different reasoning tasks in ranked ontologies in general, as well as in the lightweight DLs \mathcal{EL} and $DL\text{-Lite}_{\text{core}}$. Note that our generic complexity results can also be applied to other DLs, such as Horn $SHIQ$, or even other logics beyond DLs. All results can also be easily extended to non-Boolean variables with finite domains. All semantic results (but not the computational complexity results) also hold for non-polynomial preference rankings.

Another natural application of our approach, especially when considering data sources from the Web and Big Data, is handling trust on the obtained answers. As knowledge and data may be extracted from sources with different reputation (e.g., Wikipedia, different newspapers, or experts), a user may want to prioritize those answers that arise from the most preferred sources, potentially conditioned on additional factors (e.g., when speaking of politics or sports). Naturally, these preferences may be different for distinct users of the system.

An interesting topic for future work is to adapt specific query answering techniques to produce effective algorithms that can be used in practice, e.g., starting from the \mathcal{EL} and $DL\text{-Lite}$ families of DLs for which query answering techniques, mostly based on rewriting, have been largely studied.

Acknowledgments

This work was supported by the DFG under RoSI (GRK 1907), by the UK EPSRC grants EP/J008346/1, EP/L012-138/1, and EP/M025268/1, and by The Alan Turing Institute under the EPSRC grant EP/N510129/1.

References

- [Amor *et al.*, 2015] Nahla Ben Amor, Didier Dubois, H ela Gouider, and Henri Prade. Possibilistic conditional preference networks. In *Proc. ECSQARU*, pp. 36–46, 2015.

- [Artale *et al.*, 2009] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyashev. The *DL-Lite* family and relations. *J. Artif. Intell. Res.*, 36:1–69, 2009.
- [Baader *et al.*, 2005] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the \mathcal{EL} envelope. In *Proc. IJCAI*, 2005.
- [Baader *et al.*, 2007] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider (eds.). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2nd edition, 2007.
- [Ben Amor *et al.*, 2003] Nahla Ben Amor, Salem Benferhat, and Khaled Mellouli. Anytime propagation algorithm for min-based possibilistic graphs. *Soft Comput.*, 8(2):150–161, 2003.
- [Ben Amor *et al.*, 2014] Nahla Ben Amor, Didier Dubois, H ela Gouider, and Henri Prade. Possibilistic networks: A new setting for modeling preferences. In *Proc. SUM*, pp. 1–7, 2014.
- [Benferhat *et al.*, 1999] Salem Benferhat, Didier Dubois, Laurent Garcia, and Henri Prade. Possibilistic logic bases and possibilistic graphs. In *Proc. UAI*, pp. 57–64, 1999.
- [Benferhat *et al.*, 2001] Salem Benferhat, Didier Dubois, and Henri Prade. Towards a possibilistic logic handling of preferences. *Appl. Intell.*, 14(3):303–317, 2001.
- [Benferhat *et al.*, 2002] Salem Benferhat, Didier Dubois, Laurent Garcia, and Henri Prade. On the transformation between possibilistic logic bases and possibilistic causal networks. *Int. J. Approx. Reason.*, 29(2):135–173, 2002.
- [Borgelt and Kruse, 2003] Christian Borgelt and Rudolf Kruse. Operations and evaluation measures for learning possibilistic graphical models. *Artif. Intell.*, 148(1/2):385–418, 2003.
- [Borgwardt *et al.*, 2016] Stefan Borgwardt, Bettina Fazzinga, Thomas Lukasiewicz, Akanksha Shrivastava, and Oana Tifrea-Marcuska. Preferential query answering over the Semantic Web with possibilistic networks. In *Proc. IJCAI*, pp. 994–1000, 2016.
- [Boutillier *et al.*, 2004] Craig Boutillier, Ronen I. Brafman, Carmel Domshlak, Holger H. Hoos, and David Poole. CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *J. Artif. Intell. Res.*, 21:135–191, 2004.
- [Brini *et al.*, 2005] Asma H. Brini, Mohand Boughanem, and Didier Dubois. A model for information retrieval based on possibilistic networks. In *Proc. SPIRE*, pp. 271–282, 2005.
- [Calvanese *et al.*, 2007] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. Autom. Reasoning*, 39(3), 2007.
- [Ceylan and Pe alozza, 2015] İsmail İlkan Ceylan and Rafael Pe alozza. Probabilistic query answering in the Bayesian description logic \mathcal{BEL} . In *Proc. SUM*, pp. 1–15, 2015.
- [Ceylan and Pe alozza, 2017] İsmail İlkan Ceylan and Rafael Pe alozza. The Bayesian ontology language \mathcal{BEL} . *J. Autom. Reasoning*, 58(1):67–95, 2017.
- [Chandra and Merlin, 1977] Ashok K. Chandra and Philip M. Merlin. Optimal implementation of conjunctive queries in relational data bases. In *Proc. STOC*, pp. 77–90, 1977.
- [d’Amato *et al.*, 2008] Claudia d’Amato, Nicola Fanizzi, and Thomas Lukasiewicz. Tractable reasoning with Bayesian description logics. In *Proc. SUM*, pp. 146–159, 2008.
- [Darwiche, 2009] Adnan Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.
- [Di Noia *et al.*, 2013] Tommaso Di Noia, Thomas Lukasiewicz, and Gerardo I. Simari. Reasoning with semantic-enabled qualitative preferences. In *Proc. SUM*, pp. 374–386, 2013.
- [Di Noia *et al.*, 2015] Tommaso Di Noia, Thomas Lukasiewicz, Maria Vanina Martinez, Gerardo I. Simari, and Oana Tifrea-Marcuska. Combining existential rules with the power of CP-theories. In *Proc. IJCAI*, pp. 2918–2925, 2015.
- [Dubois and Prade, 2004] Didier Dubois and Henri Prade. Possibilistic logic: A retrospective and prospective view. *Fuzzy Sets and Systems*, 144(1):3–23, 2004.
- [Dubois *et al.*, 2006] Didier Dubois, J er me Mengin, and Henri Prade. Possibilistic uncertainty and fuzzy features in the Semantic Web: A preliminary discussion. In *Fuzzy Logic and the Semantic Web, Capturing Intelligence*, pages 101–114. Elsevier, 2006.
- [Dubois *et al.*, 2013] Didier Dubois, Henri Prade, and Fay al Touazi. Conditional preference nets and possibilistic logic. In *Proc. ECSQUARU*, pp. 181–193, 2013.
- [Giese *et al.*, 2015] Martin Giese, Ahmet Soylu, Guillermo Vega-Gorgojo, Arild Waaler, Peter Haase, Ernesto Jim enez-Ruiz, Davide Lanti, Mart n Rezk, Guohui Xiao,  zg ur L.  z ep, and Riccardo Rosati. Optique: Zooming in on Big Data. *IEEE Computer*, 48(3):60–67, 2015.
- [HadjAli *et al.*, 2011] Allel HadjAli, Souhila Kaci, and Henri Prade. Database preference queries—A possibilistic logic approach with symbolic priorities. *AMAI*, 63(3/4): 357–383, 2011.
- [Hollunder, 1995] Bernhard Hollunder. An alternative proof method for possibilistic logic and its application to terminological logics. *Int. J. Approx. Reason.*, 12(2):85–109, 1995.
- [Joachims, 2002] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proc. SIGKDD*, pp. 133–142, 2002.
- [Krentel, 1988] Mark W. Krentel. The complexity of optimization problems. *J. Comput. Syst. Sci.*, 36(3):490–509, 1988.
- [Liau and Yao, 2001] Churn-Jung Liau and Y. Y. Yao. Information retrieval by possibilistic reasoning. In *Proc. DEXA*, 2001.
- [Lukasiewicz and Schellhase, 2007] Thomas Lukasiewicz and J rg Schellhase. Variable-strength conditional preferences for ranking objects in ontologies. *J. Web Sem.*, 5(3):180–194, 2007.
- [Lukasiewicz *et al.*, 2013] Thomas Lukasiewicz, Maria V. Martinez, and Gerard I. Simari. Preference-based query answering in Datalog+/- ontologies. In *Proc. IJCAI*, pp. 501–518, 2013.
- [Lukasiewicz *et al.*, 2014] Thomas Lukasiewicz, Maria Vanina Martinez, and Gerardo I. Simari. Probabilistic preference logic networks. In *Proc. ECAI*, pp. 561–566, 2014.
- [Pearl, 1988] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. 1988.
- [Poole, 1997] David Poole. The independent choice logic for modelling multiple agents under uncertainty. *Artif. Intell.*, 94, 1997.
- [Qi *et al.*, 2011] Guilin Qi, Qiu Ji, Jeff Z. Pan, and Jianfeng Du. Extending description logics with uncertainty reasoning in possibilistic logic. *Int. J. Intell. Syst.*, 26(4):353–381, 2011.
- [Rosati, 2007] Riccardo Rosati. On conjunctive query answering in \mathcal{EL} . In *Proc. DL*, CEUR-WS.org, 2007.
- [Siberski *et al.*, 2006] Wolf Siberski, Jeff Z. Pan, and Uwe Thaden. Querying the Semantic Web with preferences. In *Proc. ISWC*, pp. 612–624, 2006.
- [Suciu *et al.*, 2011] Dan Suciu, Dan Olteanu, Christopher R e, and Christoph Koch. *Probabilistic Databases*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.