A New $DL\text{-}Lite_{bool}^{N}$ Probabilistic Extension Using Belief

Ala Djeddai, Hassina Seridi and Med Tarek Khadir

LabGED Laboratory, Computer Science Department, Badji Mokhtar-Annaba University P.O. Box 12, 23000 Annaba, Algeria.
{djeddai,seridi,khadir}@labged.net

Abstract. Dealing with uncertainty is a very important issue in description logics (DLs). In this paper, we present $PrDL\text{-}Lite_{bool}^{N}$ a new probabilistic extension of $DL\text{-}Lite_{bool}^{N}$ by supporting the belief interval in a single axiom or a set of axioms connected with conjunction (by $\wedge$) or disjunction (by $\vee$) operators. The $PrDL\text{-}Lite_{bool}^{N}$ semantics is based on $DL\text{-}Lite_{bool}^{N}$ features which are a new alternative semantics for $DL\text{-}Lite_{bool}^{N}$ having a finite structure and its number is always finite unlike classical models. $PrDL\text{-}Lite_{bool}^{N}$ also supports terminological and assertional probabilistic knowledge and the main reasoning tasks: satisfiability, deciding the probabilistic axiom entailment and computing tight interval for entailment are achieved by solving linear constraints system. A prototype is implemented using OWL API for knowledge base creation, Pellet for reasoning and LpSolve for solving the linear programs.

Keywords: uncertainty, description logics, probabilistic reasoning, DL-Lite, belief probability.

1 Introduction

Dealing with uncertainty in knowledge representation and reasoning is a very important issue and research direction. Uncertainty arises from various causes such as: automatically extracting and processing data, integration of information from different heterogeneous sources, inconsistency, incompleteness and incorrect information. Ontology merging, user or automatic annotations, ontology alignment and information retrieval are also important sources of uncertainty. An example of uncertain information is given as follows: "Roufaida is a postgraduate student with degree ≥ 0.7". From the main languages used to represent and reason about knowledge, there are description logics DLs [1] that are designed for crisp and deterministic information. Thus, they are not able to deal with the unknown and therefore must be extended in order to comply with uncertain knowledge. DLs are the formal foundation of the ontology web language OWL which is a W3C standard used for knowledge modeling in the semantic web. To handle uncertainty, many approaches proposed DLs extensions such as, probabilistic approaches when the degree of uncertainty is interpreted as probability value. Most of them are difficult to use and based on classical models or use graphical models (such as: Bayesian Network) and don't supporting the belief in terminological axioms. In this paper, $PrDL\text{-}Lite_{bool}^{N}$ a novel probabilistic extension of $DL\text{-}Lite_{bool}^{N}$ [2] by using the belief probability is presented. An example of probabilistic axiom is: a given professor is a PhD student with probability in [0.7,0.9]. We choose working with intervals instead of single values because the probabilities can be extracted from different sources and different agents can compute different probabilities so intervals are a good choice for working under uncertainty. The $PrDL\text{-}Lite_{bool}^{N}$ semantics is based on $DL\text{-}Lite_{bool}^{N}$ features which are a new alternative semantics for $DL\text{-}Lite_{bool}^{N}$. We use features instead of classical models because they have finite structure and its number is always finite unlike models. $PrDL\text{-}Lite_{bool}^{N}$ supports terminological and assertional probabilistic knowledge. Unlike approaches with graphical models when the model must be fully specified, $PrDL\text{-}Lite_{bool}^{N}$ needs only the belief interval in a single axiom or a set of axioms connected with $\wedge$ or $\vee$. Using features with belief is a new contribution compared to the work in [13] which uses probabilistic interpretation on all features but with conditional probability that are interpreted as statistical information and not belief.

Section 2 presents the $DL\text{-}Lite_{bool}^{N}$ language and the feature notion. In section 3 the proposed probabilistic extension is presented and the syntax and semantics of $PrDL\text{-}Lite_{bool}^{N}$ knowledge base based on features are explained. Section 4 details the reasoning tasks supported by $PrDL\text{-}Lite_{bool}^{N}$. The implementation and experimentation are given in section 5. The section 6 is for the related works where the conclusion and future works are presented in the last section.

2 $DL\text{-}Lite_{bool}^{N}$ Language and the Feature Notion

In this section, we start by defining $DL\text{-}Lite_{bool}^{N}$, its syntax and semantics, then the notion of types and features are detailed.

2.1 The $DL\text{-}Lite_{bool}^{N}$ Language

Description logics [1] abbreviated by DLs from a family of languages that are used for knowledge representation and reasoning. Their complexity increased with their expressivity. Therefore some researchers propose $DL\text{-}Lite$ language [3,4] with very good computational property but less expressivity. It is behind OWL 2 QL which is OWL 2 profile. Thus, we focus on $DL\text{-}Lite_{bool}^{N}$ [2] which is an expressive superset of $DL\text{-}Lite$ where the latter is extended with full Booleans and number restrictions on roles. It contains or individual, atomic concepts and atomic roles. General concepts and roles are defined as follows: $R \leftarrow P|P^{-}$, $B \leftarrow \top|A| \geq n\ R$, $C \leftarrow B|\neg C|C_1 \sqcap C_2$, where A is an atomic concept, P is an atomic role, R is a general role and $n \geq 1$. B is called basic concept and C is a general concept. We abbreviate $\neg\top$, $\geq 1\ R$, $\neg(\neg C_1 \sqcap \neg C_2)$ and $\neg(\geq n+1\ R)$ respectively by $\bot$, $\exists R$, $C_1 \sqcup C_2$ and $\leq n\ R$.

A signature is a finite set $S = S_C \cup S_R \cup S_I \cup S_N$ where S_C is the set of atomic concepts, S_R is the set of atomic roles, S_I is the set of individual names and S_N is the set of natural numbers used in S_C (1 is always in S_N). A $DL\text{-}Lite^N_{bool}$ $TBox\ T$ is a finite set of concept inclusions on the form $C_1 \sqsubseteq C_2$, where C_1 and C_2 are general concepts. A $DL\text{-}Lite^N_{bool}$ $ABox\ A$ is a finite set of assertions of the form $C(a)$ (concept membership) or $R(a,b)$ or $\neg R(a,b)$ (role membership) where a and b are individuals names. The pair (T,A) forms a $DL\text{-}Lite^N_{bool}$ knowledge base. The semantics of $DL\text{-}Lite^N_{bool}$ is given by an interpretation $I = (\Delta^I, .^I)$ where Δ^I is a non empty set called the interpretation domain and $.^I$ is an interpretation function that associates every individual a with an element $a^I \in \Delta^I$ such that $a^I \neq b^I$ for every pair $a, b \in S_I$, every atomic concept A with a subset (unary relation) A^I of Δ^I and every atomic role P with a subset (binary relation) P^I of $\Delta^I \times \Delta^I$. The interpretation I is extended to general concepts and roles. Given an interpretation I, we write I satisfies $C_1 \sqsubseteq C_2$ denoted by $I \vDash C_1 \sqsubseteq C_2$ if ${C_1}^I \subseteq {C_2}^I$, $I \vDash C(a)$ if $a^I \in C^I$, $I \vDash R(a,b)$ if $(a^I, b^I) \in R^I$. I satisfies a $TBox\ T$ if I satisfies every inclusion in T, I satisfies a $ABox\ A$ if it satisfies each assertion in A. Given a knowledge base $K = (T,A)$ and an interpretation I, I is called a model of K if I satisfies T and A. A knowledge base K is satisfiable if it has at least one model. For a concept C, we say that K satisfies C if there is a model of K satisfying C. For concept inclusion or assertion x, we say that x is entailed by K and we write $K \vDash x$ if x is satisfied by every model of K.

Terminological Box T:	Assertion Box A:
1. $PhDStudent \sqsubseteq Student$	5. $Professor$(FOFO)
2. $Course \sqsubseteq \neg Student \sqcap \neg Professor$	6. $Course$(DESCRIPTION LOGICS)
3. $\exists teach \sqsubseteq Professor$	7. $teach$(FOFO, DESCRIPTION LOGICS)
4. $\exists teach^- \sqsubseteq Course$	

Fig. 1. An example of $DL\text{-}Lite^N_{bool}$ knowledge base

Example 1. An example of $DL\text{-}Lite^N_{bool}$ knowledge base $K = \langle T, A\rangle$ is presented in fig.1 which contains the atomic concepts $Student$, $PhDStudent$, $Professor$ and $Course$. It also contains the atomic role $teach$. The TBox T tells that all PhD students are students (1) and a course is not a professor or a student. It also says that the atomic role $teach$ has $Professor$ as domain (3) and $Course$ as range (4). The ABox A defining FOFO as a professor teaches DESCRIPTION LOGICS (5) (6) (7). The signature of K is $S_C \cup S_R \cup S_I \cup S_N$ where: $S_C = \{Professor, Student, PhDStudent, Course\}$, $S_R = \{teach\}$, $S_I = \{$FOFO, DESCRIPTION LOGICS$\}$, $S_N = \{1\}$.

2.2 $DL\text{-}Lite^N_{bool}$ Features

Working with models has some difficulties. Domains have complex (possibly infinite) structures and DL knowledge bases may have infinitely many models. Thus an alternative semantics has been proposed called feature [15] which is prposed for the $DL\text{-}Lite_{bool}^{N}$ knowledge bases. In contrast to classical models the features have always finite structures and every knowledge base has a finite set of features. According to these motivations, we choose working with features instead of models. Another motivation is that we must consider all possible knowledge base situations and the number of the latters must be finite. The feature notion is based on the type notion proposed in [8]. We begin by presenting types and then the feature is explained.

Given a finite signature S, an $S\text{-}type\ \tau$ is a set of basic concepts over S, such that: $\top \in \tau$ and for any $m, n \in S_N$ with $m < n$, $R \in S_R \cup \{P^- | P \in S_R\}$, $\geq n\ R \in \tau$ implies $\geq m\ R \in \tau$. In what follows, $\top \in \tau$ is omitted for simplicity and $S\text{-}type$ will be specifies as $type$. Type τ satisfies basic concept B if $B \in \tau$, τ satisfies $\neg C$ if τ not satisfies C, and τ satisfies $C_1 \sqcap C_2$ if τ satisfies C_1 and C_2. The satisfaction relation is denoted by $\vDash$. We say that τ satisfies $C_1 \sqsubseteq C_2$ ($\tau \vDash C_1 \sqsubseteq C_2$) if $\tau \vDash \neg C_1$ or $\tau \vDash C_2$. Thus τ satisfies a $TBox\ T$ if τ satisfies every concept inclusion axiom in T.

Types are sufficient to capture the semantics of the $TBox$, they are not able to capture the semantics of the $ABox$ because they are not suited for individual, thus they must be extended with additional set. The latter is dedicated to the concept and role memberships and is called a $S\text{-}Herbrand$ set for the $ABox$, defined in [15] as:

Definition 1. An $S\text{-}Herbrand$ set H (or $Herbrand$) is finite set of assertions of the form $B(a)$ or $P(a, b)$, where $a, b \in S_I$, $P \in S_R$ and B is a basic concept over S, satisfying the following conditions:

- For each $a \in S_I$, $\top(a) \in H$, and $\geq n\ R(a) \in H$ implies $\geq m\ R(a) \in H$ for $m, n \in S_N$ with $m < n$.
- For each $P \in S_R$, $P(a, b_i) \in H$ $(i = 1, \ldots, n)$ implies $\geq m\ P(a) \in H$ for any $m \in S_N$ such that $m \leq n$.
- For each $P \in S_R$, $P(b_i, a) \in H$ $(i = 1, \ldots, n)$ implies $\geq m\ P^-(a) \in H$ for any $m \in S_N$ such that $m \leq n$.

For a given individual a, the type $\tau = \{B_1, \ldots B_j\}$ $(j \geq 1)$ is called the type of a in H, where $B_1(a), \ldots B_j(a)$ are all basic concept assertions associated with a in H. A $Herbrand$ set H satisfies $C(a)$ if the type of a in H satisfies C, H satisfies $P(a, b)$ or $P^-(b, a)$ if $P(a, b) \in H$ (the same is with $\neg P(a, b)$ and $\neg P^-(b, a)$). H satisfies an $ABox\ A$ if H satisfies every assertions in A. The pair $\langle \tau, H \rangle$ can be used to provide a semantics characterization but it is proved in [15] that using this pair is not sufficient to capture the connection between the $TBox$ and the $ABox$. Thus feature which is a set of types can provide a complete semantics of $DL\text{-}Lite_{bool}^{N}$ knowledge bases. The notion of feature is defined in [15] as follows:

Definition 2. Given a signature S, an $S\text{-}feature$ (or simply $feature$) is a pair $F = \langle \Xi, H \rangle$, where Ξ is a non empty set of $S\text{-}types$ and H an $S\text{-}Herbrand$ set, satisfying the following conditions:

- $\exists P \in \bigcup \Xi$ if $\exists P^- \in \bigcup \Xi$ for each $P \in S_R$.

- For each $a \in S_I$ we have $\tau \in \Xi$, where τ is the type of a in H.

Given a feature $F = \langle \Xi, H \rangle$, F satisfies $C_1 \sqsubseteq C_2$ if every type in Ξ satisfies $C_1 \sqsubseteq C_2$, F satisfies an assertion $C(a)$ or $P(a, b)$ if H satisfies this assertion, F satisfies a $TBox\ T$ if F satisfies every inclusion in T, F satisfies an $ABox\ A$ if F satisfies every assertion in A. Given a $DL\text{-}Lite_{bool}^{N}$ knowledge base $K = \langle T, A \rangle$ and a feature F, F is a model feature of K if F satisfies T and A. The set of all model features of K is denoted by $M_f(K)$. For a concept inclusion or assertion x, $K \vDash_f x$ if all model features of K satisfies x [15]. For further reading, the readers are referred to [8] and [15].

Example 2. Given a feature $F = \langle \Xi, H \rangle$ defined over the signature of K such that: $\Xi = \{\tau_1, \tau_2\}$ where: $\tau_1 = \{Professor, \exists teach\}$ and $\tau_2 = \{\exists teach^{-}, Course\}$, $H = \{Professor(\text{FOFO}), Course(\text{DESCRIPTION LOGICS}), teach(\text{FOFO}, \text{DESCRIPTION LOGICS})\}$.

The feature $F = \langle \Xi, H \rangle$ respects the conditions in definition 2 and the $Herbrand$ set respects the definition 1. Every individual a in S_I has a type in H, τ_1 is for FOFO and τ_2 is for DESCRIPTION LOGICS. F is model feature of K since every type satisfies every inclusion in T and H satisfies every assertion in A. Thus $F = \langle \Xi, H \rangle \in M_f(K)$.

3 Probabilistic Extension based on Features Using Belief

A novel probabilistic extension of $DL\text{-}Lite_{bool}^{N}$ based on features is here presented. The $DL\text{-}Lite_{bool}^{N}$ is extended to specify belief interval about its axioms. The extension is called $PrDL\text{-}Lite_{bool}^{N}$. In this section, the syntax and semantics of $PrDL\text{-}Lite_{bool}^{N}$ probabilistic knowledge bases using $DL\text{-}Lite_{bool}^{N}$ features are given.

3.1 Syntax of $PrDL\text{-}Lite_{bool}^{N}$ Probabilistic Knowledge Bases

The Axioms in $DL\text{-}Lite_{bool}^{N}$ knowledge bases can be annotated by belief degree interval. The following types of probabilistic axioms are supported:

1. Probabilistic terminological axioms ($TBox$ axioms): probabilistic concept inclusions (PCI for short) about relationship between concepts. Each one has the form $(C \sqsubseteq D)_{[\alpha,\beta]}$ which signifies that we have a belief degree in $[\alpha, \beta]$ that the concept D is subsumed by C or C is sub class of D.
2. Probabilistic $ABox$ axioms: probabilistic assertions about concepts and roles instances: $C(a)_{[\alpha,\beta]}$ means that we have a belief degree in $[\alpha, \beta]$ that the individual a is an instance of the concept C. $R(a, b)_{[\alpha,\beta]}$ means that the individual a is related with the individual b by the role R with a belief degree in $[\alpha, \beta]$.
3. Using conjunction ($\wedge$) or disjunction ($\vee$) are not allowed in DLs, thus the satisfaction of axioms that contain these notations is not defined for features. Therefore we define it as follow: for a given axiom $x = x_1 \wedge x_2 \ldots \wedge x_n$ where each x_i is a $DL\text{-}Lite_{bool}^{N}$ axiom, we say that a feature $F = \langle \Xi, H \rangle$ satisfied x if $F \vDash x_i$ for every x_i in x and we write

$F \vDash x$. We say that F satisfies $x = x_1 \vee x_2 \ldots \vee x_n$ if F satisfies at least one x_i. Belief about these axioms is allowed in $PrDL\text{-}Lite^N_{bool}$. Thus, the probabilistic axiom $x = (x_1 \wedge x_2 \ldots \wedge x_n)_{[\alpha,\beta]}$ means that we have a belief in $[\alpha, \beta]$ that all x_i can be satisfied in the same situation. The probabilistic axiom $(x_1 \vee x_2 \ldots \vee x_n)_{[\alpha,\beta]}$ means that we have a belief in $[\alpha, \beta]$ that at least one x_i can be satisfied. This type of axioms is called probabilistic conjunction and disjunction axioms ($PCDA$ for short). Using $\wedge$ is not allowed with $\vee$ in the same $PCDA$ axiom

The values α and β are in [0,1] where $\alpha \leq \beta$, α is the lower bound and β is the upper bound. In $PrDL\text{-}Lite^N_{bool}$, the probabilistic terminological box PT is a finite set of PCIs. The probabilistic assertions box PA is a finite set of probabilistic assertions. $PCDA$ is a set of conjunction and disjunction axioms. A probabilistic knowledge base KB in $PrDL\text{-}Lite^N_{bool}$ is defined as $KB = \langle T, PT, A, PA, PCDA \rangle$, the axioms of $T \cup A$ are called certain axioms whereas the axioms in $PT \cup PA \cup PCDA$ are uncertain axioms. Probabilistic Axiom with [1,1] are considered as certain axiom. Thus they are removed and added to $T \cup A$. A single belief value is allowed, thus in this case $\alpha = \beta$ (see axiom 10 in fig.2).

To model a $PrDL\text{-}Lite^N_{bool}$ probabilistic knowledge, we must have an ontology (T and A) in $DL\text{-}Lite^N_{bool}$ and then extend it by adding probabilistic axioms.

Example 3. We present an example of probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA \rangle$ in $PrDL\text{-}Lite^N_{bool}$ which is an extension of $K = (T, A)$ in fig.1 with additional probabilistic axioms. Axioms numbered from 1 to 7 are the same in K. The PTBox PT tells that the concept professor is a subclass of the concept PhD student with a belief degree in [0.44,0.65] (8). The PA defining RIDA and KAMEL as respectively a PhD student (9) and a professor (10) with belief degree respectively in [0.55,0.60] and [0.80,0.80]. The $PCDA$ axiom 11 says that we have a belief degree in [0.45,0.67] that LOTFI is a professor and teacher of DISCRIPTION LOGICS. Axiom 12 tell that ALA is a PhD student or professor with a belief degree in [0.30,0.60].

Probabilistic Terminological Box PT:	Probabilistic Assertion Box PA:
8. $(Professor \sqsubseteq PhDStudent)_{[0.44,0.65]}$	9. $PhdStudent(\text{RIDA})_{[0.55,0.60]}$ 10. $Professor(\text{KAMEL})_{[0.80,0.80]}$
Probabilistic Conjunction and Disjunction Axioms $PCDA$: 11. $(Professor(\text{LOTFI}) \wedge teach(\text{LOTFI}, \text{DESCRIPTION LOGICS}))_{[0.45,0.67]}$ 12. $(PhDStudent(\text{ALA}) \vee Professor(\text{ALA}))_{[0.30,0.60]}$	

Fig. 2. An example of $PrDL\text{-}Lite^N_{bool}$ probabilistic knowledge base

3.2 Semantics of $PrDL\text{-}Lite^N_{bool}$ Probabilistic Knowledge Bases

Given a probabilistic axiom, the certain axiom is specified by removing the belief interval. The certain one of $(C \sqsubseteq D)_{[\alpha,\beta]}$ denoted by $cer((C \sqsubseteq D)_{[\alpha,\beta]})$ is $C \sqsubseteq D$. Given a set of PCIs PT, the certain set of PT denoted by $cer(PT)$ is a set of concept inclusions. Probabilistic assertion and the set of probabilistic assertions PA are treated in the same manner. Because every axiom in $PCDA$ has at least two axioms ($\vee$ or $\wedge$ must connects more than an axiom), a set of certain axioms can be extracted from every $PCDA$ axiom, this set can includes concept inclusions and assertions. Therefore two functions: cer_c and cer_d are defined to extract these sets where $cer\_c((x_1 \wedge x_2 \ldots \wedge x_n)_\alpha)$ is a set contains all $cer(x_i)$ (all x_i must be satisfied), and $cer\_d((x_1 \vee x_2 \ldots \vee x_n)_\alpha)$ is a set contains at least a $cer(x_i)$ (at least one x_i must be satisfied). Thus $cer(PCDA)$ is a set contains all sets of certain axioms of all $PCDA$ axioms.

A set of deterministic knowledge bases can be extracted from KB, everyone must contains $T \cup A$ and it can contains selected axioms from $cer(PT) \cup cer(PA)$ and selected sets of axioms from $cer(PCDA)$. Axioms from $cer(PT) \cup cer(PA)$ are added directly. For every selected set from $cer(PCDA)$, all its certain axioms are added. Each deterministic knowledge base respects the $DL\text{-}Lite^{N}_{bool}$ language and every satisfiable knowledge base is called $possible\ knowledge\ base$.

Definition 3 ($\boldsymbol{possible\ knowledge\ bases}$). Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA \rangle$. The possible knowledge bases of KB, denoted by $PosKB$ is defined as follows: $PosKB = \{K = \langle T \cup T', A \cup A' \rangle | K$ is satisfiable where T' (resp. A') contains selected axioms from $cer(PT)$ (resp. $cer(PA)$) and concept inclusions (resp. assertions) in the selected certain sets from $cer(PCDA)\}$.

In other words, the possible knowledge base K is a possible consistent situation of KB. In this situation, the actual world is a model of K. The satisfiability condition is important for preventing inconsistencies and contradictions that may occur between the axioms in $cer(PT)$, $cer(PA)$ and $cer(PCDA)$. If there are probabilistic axioms that have certain axioms which are inconsistent with $T \cup A$ then they can't participated in creating $PosKB$, thus they are omitted and do not considered in reasoning.

Using definition 3, another notion is defined which is $the\ possible\ feature$:

Definition 4 ($\boldsymbol{possible\ features}$). Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA \rangle$, let $PosKB$ be the set of all possible knowledge bases of KB. The possible features related to KB, denoted by $Pos\mathcal{F}$ is defined as follows: $Pos\mathcal{F} = \{F | F\ is\ a\ model\ feature\ of\ K\ such\ that\ K \in PosKB\}$.

From the definition, $Pos\mathcal{F}$ contains all model features of all possible knowledge bases in $PosKB$. Like models, the possible features are used to describe the current KB situation. The signature of probabilistic knowledge base KB is defined as a finite set $S = sig(T) \cup sig(cer(PT)) \cup sig(A) \cup sig(cer(PA)) \cup sig(cert(PDCA))$.

The semantics of $PrDL\text{-}Lite^{N}_{bool}$ probabilistic knowledge bases is given by probabilistic interpretations. A probabilistic interpretation Pr is a probability function on all possible features in $Pos\mathcal{F}$ (Pr: $Pos\mathcal{F} \rightarrow [0,1]$) such the sum of all $Pr(F)$ is equal to 1. We have a probability distribution over $Pos\mathcal{F}$ and Pr distributed its probability values only on possible features i.e. considering only possible situations of KB. The probability of any axiom x is the sum of the probabilities associated with all features that satisfy x: $Pr(x) =$

$\sum_{F \in Pos\mathcal{F} \text{ and } F \vDash x} Pr(F)$. A probabilistic interpretation Pr satisfies a PCI $(C \sqsubseteq D)_{[\alpha,\beta]}$ denoted by $Pr \vDash (C \sqsubseteq D)_{[\alpha,\beta]}$ if and only if $Pr(C \sqsubseteq D) \in [\alpha, \beta]$. Pr satisfies a set of PCIs PT denoted by $Pr \vDash PT$ if Pr satisfies all elements in PT. The same is with a probabilistic assertion and a set of probabilistic assertions PA. Pr satisfies $PDCA$ denoted by $Pr \vDash PDCA$ if Pr satisfies all elements in $PCDA$. Pr satisfies a certain concept inclusion $C \sqsubseteq D$, denoted by $Pr \vDash C \sqsubseteq D$ if and only if for every feature with $Pr(F) > 0$, we have $F \vDash C \sqsubseteq D$. Pr satisfies a set of concept inclusions T denoted by $Pr \vDash T$ if Pr satisfies all elements in T. The same is with certain assertion and certain set of assertions A. Therefore, a probabilistic interpretation Pr is a model of probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA\rangle$ if and only if Pr satisfies T, A, PT, PA and $PDCA$. KB is satisfiable (or consistent) if there is at least a model of KB. Given a probabilistic axiom $x_{[\alpha,\beta]}$, KB entails $x_{[\alpha,\beta]}$, denoted by $KB \vDash x_{[\alpha,\beta]}$ if for every Pr such that $Pr \vDash KB$ we have $Pr \vDash x_{[\alpha,\beta]}$. Before checking the satisfiability of KB, $\langle T, A\rangle$ must be consistent.

Example 4. The KB in fig.2 has the signature $S = S_C \cup S_R \cup S_I \cup S_N$ where: S_C, S_R, and S_N are the same of K in fog.1 but $S_I = \{\text{FOFO}, \text{ALA}, \text{LOTFI}, \text{RIDA}, \text{KAMEL}\}$. The set $PosKB$ of KBcontains for example $K_1 = \langle T \cup \{cer(8)\}, A \cup \{cer(9), cer(10)\}\rangle$. We observe that K_1 is satisfiable and it respects the expressiveness of $DL\text{-}Lite_{bool}^{N}$.

4 Reasoning and Inferences Tasks

The main reasoning and inference tasks for $PrDL\text{-}Lite_{bool}^{N}$ are the following:

- Probabilistic Knowledge Base Satisfiability ($PKBSAT$): Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA\rangle$, decide whether KB is satisfiable.
- Tightest Belief interval for Logical Entailment ($TBILogEn$): Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA\rangle$ and an axiom x, compute the tightest belief interval $[\alpha, \beta]$ such that $KB \vDash x_{[\alpha,\beta]}$.
- Logical Entailment ($LogEn$): Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA\rangle$ and probabilistic axiom $x_{[\alpha,\beta]}$ associated with belief interval $[\alpha, \beta]$, decide whether $KB \vDash x_{[\alpha,\beta]}$ or not.

The first task is achieved by using theorem 1. The second and the thirds uses theorem 2. Similarly to [9], the reasoning tasks use a system of linear constraints.

Theorem 1. Let $KB = \langle T, PT, A, PA, PCDA\rangle$ a probabilistic KB. This latter is satisfiable if the next linear constraints system over variables p_F ($F \in pos\mathcal{F}$) is solvable:

$$\sum_{F \in pos\mathcal{F}, F \vDash x} p_F \geq \alpha \;\; (one\ for\ each\ x_{[\alpha,\beta]}\ in\ PT\ and\ PA\ and\ PCDA)$$
$$\sum_{F \in pos\mathcal{F}, F \vDash x} p_F \leq \beta \;\; (one\ for\ each\ x_{[\alpha,\beta]}\ in\ PT\ and\ PA\ and\ PCDA)$$
$$\sum_{F \in pos\mathcal{F}} p_F = 1$$

$p_F \geq 0 \; for \; all \; F \in pos\mathcal{F}$

Proof. The solver tries to find assignments to every p_F such that all constraints are satisfied. Every p_F is considered as probability of $F \in pos\mathcal{F}$. Thus the solver tries to find a probability function on $pos\mathcal{F}$ that respects the constraints. The two first constraints are to respect the satisfiability of every probabilistic axiom in KB i.e., the sum of probabilities of the features that satisfy x is in $[\alpha, \beta]$. The third one is to respect that the sum of all probabilities associated with all features is 1. The condition of positive probabilities is specified in the last constraint. By respecting the two last constraints, the condition that the probability $\in [0,1]$ is also respected. If the solver get a solution i.e., value for every p_F then the solution respects the satisfiability conditions and KB has a model Pr that assigns for every $F \in pos\mathcal{F}$ a value p_F. Thus KB is satisfiable. We can proof by the same manner that if KB is satifiable then the previous system of linear constraints is solvable.

Theorem 2. Given a probabilistic knowledge base $KB = \langle T, PT, A, PA, PCDA \rangle$. Suppose KB is satisfiable. Let x be an axiom. The values α and β such that $KB \vDash x_{[\alpha,\beta]}$ are taken over all possible solutions of the system in Theorem 1 as follow:

$$\alpha = min \sum\nolimits_{F \in pos\mathcal{F}, F \vDash x} p_F \quad subject \; to \; system \; in \; Theorem \; 1$$

$$\beta = max \sum\nolimits_{F \in pos\mathcal{F}, F \vDash x} p_F \quad subject \; to \; system \; in \; Theorem \; 1$$

Proof. Suppose that KB is satisfiable, the system in Theorem 1 has at least one solution i.e., probabilistic interpretation of KB. Given an axiom x, for every solution i.e., probabilistic interpretation, the solver computes the sum of all values assigned with features that satisfy x. To minimization, it keeps α and β for the maximization. Thus, in every interpretation Pr we have $Pr(x) \in [\alpha, \beta]$ so $KB \vDash x_{[\alpha,\beta]}$.

The interval computed by theorem 2 is called the tightest belief interval i.e., α (resp., β) is the min (resp., max) of $Pr(x)$ subject to all models Pr of KB. Theorem 2 can be used to decide if a given probabilistic axiom is entailed by a satisfiable probabilistic KB. Thus $KB \vDash x_{[\alpha,\beta]}$ if the tightest belief interval that x is entailed by KB is in $[\alpha, \beta]$.

Example 5. From the example 3, if KB is satisfiable using Theorem 1, then we can use theorem 2 to compute $TBILogEn$ of some given axioms such as: computing the tightest belief interval $[\alpha, \beta]$ such that $KB \vDash PhDStudent(\text{KAMEL})_{[\alpha,\beta]}$, finding the $TBILogEn$of $PhDStudent \sqsubseteq Professor$. We can also decide whether $PhDStudent(\text{LOTFI})_{[0.4,0.6]}$ is a logical consequence of KB.

5 Implementation and Experimentation

A prototype of $PrDL\text{-}Lite_{bool}^{N}$ is implemented in Java with Eclipse, using Pellet [18] for reasoning, owlapi [19] for knowledge base creation, and LpSolve 5.5 [20] for solving the linear programs. $DL\text{-}Lite_{bool}^{N}$ supportes only concept inclusion, concept and role assertions. In owlapi the previous axioms are respectively created by OWL.$subClassOf$, OWL. $classAssertion$, OWL. $propertyAssertion$. For example $PhDStudent \sqsubseteq Student$ is represented by OWL.$subClassOf(PhDStudent, Student)$. Therefore T, A, PT, PA and $PCDA$ are created using owlapi where every probabilistic axiom is associated with its belief interval.

During the building of the possible knowledge bases, Pellet reasoner is used to test the satisfiability of the latters. For a given probabilistic KB, we can generate $2^{|PT|+|PA|+|PCDA|}$ knowledge base, thus the one in fig.2 has $2^5 = 32$ knowledge bases where all of the latters are consistent so $|PosKB| = 32$ (tested by the $PrDL\text{-}Lite_{bool}^{N}$ prototype). We conclude that in worst cases, we have $|PosKB| \leq 2^{|PT|+|PA|+|PCDA|}$. For every K in $PosKB$, using its signature $sig(K)$, we compute all types which satisfy its TBox, everyone is a set of basic concepts represented using owlapi by a set of OWLClassExpression. From the ABox of K, all basic concept and basic role assertions are extracted using Pellet. Thus these assertions are used create the $Herbrand$ set H of K according to the definition 1 (H is created in owlapi as owl ontology). All possible combinations of the types associated with K are generated. For every individual in $sig(K)$, its type in H is extracted and added to every combination. The latter is tested if it forms with H a feature of K using. Every element in $PosKB$ is treated with the same manner and all features are added to $PosF$.

	$PrDL\text{-}Lite_{bool}^{N}$	Probabilistic DL-Lite in [13]
Semantics based on features	Supported. Probabilistic interpretation on only possible features.	Supported. Probabilistic interpretation on all features of a given signature.
Terminological probabilistic knowledge	Supports the belief in concept inclusions.	Supports the conditional constraints.
Probabilistic Assertions	Supported	Supported
Conjunction axioms	supported	Not supported
Disjunction axioms	Supported	Not supported
Reasoning tasks	Strong conclusions by computing the tightest belief interval. Deciding entailment is supported. The reasoning tasks use linear programming for efficient computation.	Weak conclusions. It uses an inference rules that get only lower bounds. In most cases, the upper bound is 1. The inferences cover only special cases. The tightest interval is not supported
Implementation and evaluation	A prototype is implemented and evaluated	No implementation and evaluation

Table 1. Comparison between $PrDL\text{-}Lite_{bool}^{N}$ and the work in [13]

One feature can model more than one knowledge base, thus the number of all possible features can be reduced. Using the prototype, we found that the probabilistic KB in fig.2 has 1548 possible features and this proved that the number of features is finite. The linear program Lp in theorem 1 is created using LpSolve, the variables number is $|PosF|$ because every $F \in PosF$ is associated with one variable p_F. Using PT, PA, $PCDA$ and $PosF$, two constraints are created for every axiom x, one for its interval lower bound and one for the upper bound. The coefficient of every variable p_F in these constraints can be 1 (F satisfies x) or 0 (F does not satisfy x). The Lp for the KB in fig.2 has 1548 variables and 12 constraints (10 for the probabilistic axioms).

Example 6. Using the prototype and the probabilistic KB in fig.2, we have found that: $KB \vDash PhDStudent(\text{KAMEL})_{[0.24,0.65]}$, $(PhDStudent \sqsubseteq Professor)_{[0.0,0.45]}$ and $PhDStudent(\text{LOTFI})_{[0.40,0.60]}$ is not entailed by KB because the tightest belief interval of $PhDStudent(\text{LOTFI})$ is $[0.45,0.67]$.

For comparison, our work and the one in [13] are used (see table 1) because the only probabilistic extension of $DL\text{-}Lite^{N}_{bool}$ which based on features is in [13]. The goal of comparison is to understand the points of difference between the two works. We have not presented this section in detail because of the limitation in paper length.

6 Related Works

Closest to our work, we have [13] where the $DL\text{-}Lite^{N}_{bool}$ is extended to use conditional constraints on concepts i.e., statistical information about concepts, it supports probabilistic terminological knowledge and probabilistic assertions about concepts and roles. The probabilistic interpretation is defined on the set of all features of a given signature, unlike our work which uses only the possible features instead of all features. In [13], no reasoning tasks and implementation are presented and the belief in concept inclusion is not supported. Its inferences are weak and consider only special cases. PrDLs [14] is a probabilistic description logic which supports the belief in DL axioms. From the probabilistic knowledge, similarly to our approach, the last reference extracts a set of certain knowledge bases but using a discrete probability distribution on all possible worlds. The author in [9] who proposed P-SHOIN(D), P-SHIQ(D) and P-DL-Lite that are probabilistic extensions of the DL-SHOIN(D), SHIF(D) and DL-Lite respectively, he uses the lexicographic entailment and his work is based on Nilsson's probabilistic logic [11]. He defines a probabilistic interpretation on possible objects (everyone contains concepts that are free of probabilistic individuals), one for terminological probabilistic knowledge and one for every probabilistic individual. PRONTO reasoner [6] is based on the works in [9]. Unlike our work, [9] does not allow probabilistic role assertions and it uses a separation between probabilistic interpretations for the individuals and this makes difficulties to draw conclusions about relations between individuals. Another work is in [22] where probabilistic DLs based on the DL-ALC are presented. The work does not allow probabilistic terminological knowledge and its semantics is subjective by considering the probabilities as believe degrees. The authors use probability distributions on possible worlds where everyone is associated with a FOL interpretation. The quasi model is defined to check the consistency of the probabilistic KB. This model shares some similarities with the feature because the former is a pair of two sets of types one contains ABox types and the other contains types for the individual. Contrary to this model the feature includes TBox and ABox types and this is important to capture the KB semantics. The authors in [22] use linear constraint systems that help to construct the worlds. In contrast, our work uses only one linear constraint system after the features construction. The work in [16] allows for epistemic and statistical probabilistic annotations in DL axioms by transforming the annotated axioms to predicate logics. The authors consider the epistemic probability as belief degree. The BUNDLE [17] is a reasoner for the work in [16]. Other set of probabilistic DLs use graphical models such as Bayesian network BN as underlying probabilistic formalisms, some of these works are [7] and [5]. P-CLASSIC [7] is a probabilistic DL based on BN that supports terminological probabilistic knowledge about concepts and roles but does not allow assertional knowledge about concepts and roles. The work in [5] is an extension of DL-Lite that uses BN towards tractable probabilistic DL. For further reading about probabilistic uncertainty in semantic web, readers are referred to [10], [12] and [21].

7 Conclusion and Future Works

In this paper, $PrDL\text{-}Lite_{bool}^{N}$ a novel probabilistic extension for $DL\text{-}Lite_{bool}^{N}$ knowledge bases is presented. The proposed work allows belief interval in a single $DL\text{-}Lite_{bool}^{N}$ axiom or a set of $DL\text{-}Lite_{bool}^{N}$ axioms that are connected with $\wedge$ or $\vee$. Its semantics is based on $DL\text{-}Lite_{bool}^{N}$ features. Both terminological and assertion probabilistic knowledge are supported. Using this work, meaningful conclusions can be drawn from the probabilistic knowledge. Analysing the computational complexity of our work and implementing efficient reasoner are the main future work directions. Another future work consists in using specific application domains such as: medical.

References

1. Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., Patel-Schneider, P. F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003).
2. Artale, A., Calvanese, D., Kontchakov, R., Zakharyaschev, M.: The DL-Lite Family and Relations. J. Artif. Intell. Res. (JAIR) 36: 1-69 (2009).
3. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R. : DL-lite: Tractable description logics for ontologies. In: Proceedings of the 20th AAAI Conference on Artificial Intelligence, AAAI 2005, pp. 602-607 (2005)
4. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-lite family. Journal of Automated Reasoning, 39(3), 385-429 (2007).

5. d'Amato, C., Fanizzi, N., & Lukasiewicz, T.: Tractable reasoning with Bayesian description logics. In: International Conference on Scalable Uncertainty Management. LNCS, vol. 5291, pp. 146-159. Springer (2008).
6. Klinov, P.: Pronto: A Practical Probabilistic Description Logic Reasoner. In proceedings of the International Workshop on Uncertainty in Description Logics, vol. 613. CEUR-WS.org (2010).
7. Koller, D., Levy, A., Pfeffer, A. P-CLASSIC: A tractable probabilistic description logic. In Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference, pp. 390-397 . AAAI Press / The MIT Press (1997).
8. Kontchakov, R., Wolter, F., Zakharyaschev, M.: Can you tell the difference between DL-lite ontologies? In: Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning, KR, pp. 285-295 (2008)
9. Lukasiewicz, T.: Expressive probabilistic description logics. Artificial Intelligence journal. 172(6-7), 852-883 (2008).
10. Lukasiewicz, T., Straccia, U. Managing uncertainty and vagueness in description logics for the semantic web. Journal of Web Semantics, 6(4), pp. 291-308 (2008)
11. Nilsson, N. J.: Probabilistic logic. Artificial Intelligence journal, 28(1). p. 71-87 (1986)
12. Predoiu, L.: Probabilistic Models for the Semantic Web: A Survey. The Semantic Web for Knowledge and Data Management, pp. 74-105. IGI global (2009)
13. Ramachandran, R., Qi G., Wang, K., Wang, J. Thornton, J.: Probabilistic Reasoning in DL-Lite. In Proceedings of The Pacific Rim International Conference on Artificial Intelligence, PRICAI, pp. 480-491, Springer (2012).
14. Tao, J., Wen, Z., Hanpin, W., Lifu, W.: PrDLs: A new kind of probabilistic description logics about belief. In Okuno, H. G., Ali, M. (eds.) Proceedings of the 20th international conference on Industrial, engineering, and other applications of applied intelligent systems, IEA/AIE'07, pp. 644-654. Springer-Verlag Berlin Heidelberg (2007)
15. Wang, Z., Wang, K., Topor, R.W.: A new approach to knowledge base revision in DL-lite. In Fox, M., Poole D., (eds.) Proceedings of the 24th AAAI Conference on Artificial Intelligence, pp. 369-374. AAAI Press (2010).
16. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R.: Epistemic and Statistical Probabilistic Ontologies. Proceedings of the International Workshop on Uncertainty Reasoning for the Semantic Web, URSW, pp. 3-14, CEUR-WS.org (2012).
17. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R.: BUNDLE: A Reasoner for Probabilistic Ontologies. In Faber, W., Lembo, D., (ed.), Web Reasoning and Rule Systems: 7th International Conference, Vol. 7994, pp. 183-197. Springer (2013).
18. Sirin, E., Parsia, B., Grau, B., Kalyanpur, A., and Katz, Y. Pellet: A Practical OWL-DL Reasoner. Journal of Web Semantics 5(2), pp. 51-53 (2007).
19. OWL API, http://owlapi.sourceforge.net/.
20. Lp solve (linear programming solver), http://lpsolve.sourceforge.net/.
21. The Ninth International Workshop on Uncertainty Reasoning for the Semantic web, http://c4i.gmu.edu/ursw/2013/.
22. Lutz, C., and Schröder, L: Probabilistic Description Logics for Subjective Uncertainty. In Lin, F., Sattler, U., and Truszczynski, M., (eds.), Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning, KR, AAAI Press, (2010).