Representing Uncertain Concepts in Rough Description Logics via Contextual Indiscernibility Relations

Claudia d'Amato[1], Nicola Fanizzi[1],
Floriana Esposito[1], and Thomas Lukasiewicz[2]

[1] LACAM, Dipartimento di Informatica, Università degli Studi di Bari, Italy
firstname.lastname@uniba.it
[2] Department of Computer Science, University of Oxford, UK
thomas.lukasiewicz@cs.ox.ac.uk

Abstract. We investigate the modeling of uncertain concepts via *rough description logics (RDLs)*, which are an extension of traditional description logics (DLs) by a mechanism to handle approximate concept definitions via lower and upper approximations of concepts based on a rough-set semantics. This allows to apply RDLs to modeling uncertain knowledge. Since these approximations are ultimately grounded on an indiscernibility relation, we explore possible logical and numerical ways for defining such relations based on the considered knowledge. In particular, we introduce the notion of context, allowing for the definition of specific equivalence relations, which are directly used for lower and upper approximations of concepts. The notion of context also allows for defining similarity measures, which are used for introducing a notion of tolerance in the indiscernibility. Finally, we describe several learning problems in our RDL framework.

1 Introduction

Uncertainty is an intrinsic characteristic of the current Web, which, being a heterogeneous and distributed source of information, naturally contains uncertain as well as incomplete and/or contradictory information. Managing uncertainty is thus a highly important topic also for the extension of the Web to the Semantic Web (SW).

Particularly, modeling uncertain concepts in description logics (DLs) [1] is generally done via numerical approaches, such as probabilistic and possibilistic ones [17]. A drawback of these approaches is that uncertainty is introduced in the model (e.g., by specifying a set of uncertainty measures, such as probability and possibility measures, respectively), which often has the consequence that the approach becomes conceptually and/or computationally more complex. An alternative (simpler) approach is based on the theory of *rough sets* [22], which gives rise to new representations and *ad hoc* reasoning procedures [4]. These languages are based on the idea of *indiscernibility*.

Among these recent developments, *rough description logics* (RDLs) [23] have introduced a complementary mechanism that allows for modeling uncertain knowledge by means of crisp approximations of concepts. RDLs extend classical DLs with two modal-like operators, the lower and the upper approximation. In the spirit of rough-set theory, two concepts approximate an underspecified (uncertain) concept C as particular

F. Bobillo et al. (Eds.): URSW 2008-2010/UniDL 2010, LNAI 7123, pp. 300–314, 2013.

sub- and superconcepts, describing which elements are definitely and possibly elements of the concept, respectively.

The approximations are based on capturing uncertainty as an indiscernibility relation R among individuals, and then formally defining the upper approximation of a concept C as the set of individuals that are indiscernible from at least one that is known to belong to the concept (where $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is a standard first-order interpretation):

$$(\overline{C})^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \exists b : (a, b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}\,.$$

Similarly, one can define the lower approximation as

$$(\underline{C})^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \forall b : (a, b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}\,.$$

Intuitively, the upper approximation of a concept C covers the elements of a domain with the typical properties of C, whereas the lower approximation contains the prototypical elements of C. This may be described in terms of necessity and possibility.

To avoid introducing uncertainty into the model (as for the approaches previously mentioned), these approximations are to be defined in a crisp way. In [23], a methodology for representing approximations in a crisp way is introduced and it is also shown that RDLs can be simulated within standard DLs. Specifically, for any DL $\mathcal{DL}$ with universal and existential quantification, and symmetric, transitive and reflexive roles, the rough extension of $\mathcal{DL}$ can be translated into $\mathcal{DL}$, and reasoning in the rough extension of $\mathcal{DL}$ can be performed by reduction to $\mathcal{DL}$, using a standard DL reasoner.

However, as shown in [19, 20], the representation of the upper and the lower approximation of a concept C as crisp concepts may not be straightforward. A knowledge engineer or domain expert may not always be able to give intensional definitions of the approximated concepts, but only examples for such approximated concepts. To cope with these issues, the problem of representing concept approximations as crisp concepts can be seen as a learning problem, where one has a given set of examples (and counterexamples) for the lower (resp., upper) approximation of a given concept C, and the goal is to learn a crisp concept definition such that the examples and counterexamples are instances of the learned concept and its negation, respectively.

Looking at the semantics of the lower and upper approximations of a concept C (reported above), an important role is played by the indiscernibility relation. But to our knowledge, there are no existing works (different from [10] of which this paper is an extension) coping with the problem of defining an indiscernibility relation. Inspired by existing works on semantic metrics [2] and kernels [9], we propose to exploit semantic similarity measures, which can be optimized to maximize their capacity of distinguishing really different individuals, as indiscernibility relations. This naturally induces ways for defining an equivalence relation based on indiscernibility criteria.

The rest of this paper is organized as follows. Section 2 provides some preliminaries around DLs and RDLs. In Section 3, we introduce contextual indiscernibility relations. Section 4 proposes a family of similarity measures based on such contexts along with a suggestion on their optimization. This also allows for the definition of tolerance degrees of indiscernibility. In Section 5, we introduce and discuss the problem of learning crisp descriptions of rough concepts. Section 6 finally summarizes the results of this paper and outlines further applications of ontology mining methods.

2 Preliminaries

In this section, we first recall the basic notions of description logics (DLs). We then describe the extension of DLs to rough DLs (RDLs).

2.1 Description Logics

We now briefly recall the syntax and the semantics of DLs. For ease of presentation, we consider only the DL $\mathcal{ALC}$; for further background and details on other DLs, we refer the reader to the standard textbook [1].

Basic elements of DLs are atomic concepts and roles. *Atomic concepts* from a set $N_C = \{C, D, \ldots\}$ are interpreted as subsets of a domain of objects (resources), while *atomic roles* from a set $N_R = \{R, S, \ldots\}$ are interpreted as binary relations on such a domain (properties). *Individuals* represent the objects through names chosen from a set $N_I = \{a, b, \ldots\}$. *Complex concepts* are built using atomic concepts and roles by means of specific concept constructors. The meaning of concepts and roles is defined by *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a set of objects, called *domain*, and $\cdot^{\mathcal{I}}$ is an *interpretation function*, mapping concepts and roles to subsets of the domain and to binary relations on the domain, respectively.

The *top* concept $\top$ is interpreted as the whole domain $\Delta^{\mathcal{I}}$, while the *bottom* concept $\bot$ corresponds to $\emptyset$. Complex concepts can be built in $\mathcal{ALC}$ using the following constructors. The *conjunction* of two concepts C and D, denoted $C \sqcap D$, is interpreted as $C^{\mathcal{I}} \cap D^{\mathcal{I}}$, while the *disjunction* of C and D, denoted $C \sqcup D$, is interpreted as $C^{\mathcal{I}} \cup D^{\mathcal{I}}$. Finally, there are two restrictions on roles, namely, the *existential restriction* on R relative to C, denoted $\exists R.C$, which is interpreted as the set $\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} \colon (x, y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$, and the *value restriction* on R relative to C, denoted $\forall R.C$, which is interpreted as $\{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}} \colon (x, y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$.

More expressive DLs allow for further constructors. The DL standing behind the ontology language OWL DL is $\mathcal{SHOIQ}(\mathbf{D})$, which extends $\mathcal{ALC}$ by transitive roles, role hierarchies, nominals, inverse roles, and qualified number restrictions, and which allows to deal with concrete domains **D** and their specific semantics.

A *knowledge base* $KB = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. The TBox $\mathcal{T}$ is a set of *subsumption axioms* $C \sqsubseteq D$ and *definition axioms* $A \equiv D$, where A is usually an atomic concept, and C and D are concepts. They are *satisfied* in an interpretation $\mathcal{I}$, or $\mathcal{I}$ is a *model* of them, denoted $\mathcal{I} \models C \sqsubseteq D$ and $\mathcal{I} \models A \equiv D$, respectively, iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ and $A^{\mathcal{I}} = D^{\mathcal{I}}$, respectively. The ABox $\mathcal{A}$ contains *concept membership axioms* $C(a)$ and *role membership axioms* $R(a, b)$, where C is a concept, R is a role, and a and b are individuals. They are *satisfied* in $\mathcal{I}$, or $\mathcal{I}$ is a *model* of them, denoted $\mathcal{I} \models C(a)$ and $\mathcal{I} \models R(a, b)$, respectively, iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$, respectively. An interpretation $\mathcal{I}$ *satisfies* a knowledge base KB, or $\mathcal{I}$ is a *model* of KB, denoted $\mathcal{I} \models KB$, iff $\mathcal{I}$ satisfies all the axioms in KB. An axiom F is a *logical consequence* of KB, denoted $KB \models F$, iff every model of KB is also a model of F.

In DLs, one generally does not make the *unique name assumption (UNA)*, i.e., different individuals (which ultimately correspond to URIs in RDF/OWL) may be mapped to the same object (resource), if not explicitly forbidden. Furthermore, one usually adopts the *open-world assumption* (OWA). Thus, an object that cannot be proved to belong to a certain concept is not necessarily a counterexample for that concept. This is only interpreted as a case of insufficient (incomplete) knowledge for that assertion (i.e., models can be constructed for both the membership and non-membership case). This assumption is compatible with the typical scenario related to the Semantic Web, where new resources may continuously be made available (and unavailable) across the Web, and thus one generally cannot assume complete knowledge.

Some important inference problems in the context of DLs include subsumption checking, instance checking, and concept retrieval:

Subsumption Checking: Given a knowledge base KB and two concepts (or two roles, when role hierarchies are allowed) C and D, decide whether $KB \models C \sqsubseteq D$.

Instance Checking: Given a knowledge base KB, a concept C, and an individual a, decide whether $KB \models C(a)$.

Concept Retrieval: Given a knowledge base $KB = (\mathcal{T}, \mathcal{A})$ and a concept C, compute the set of all individuals $a \in \mathsf{Ind}(\mathcal{A})$ (among those in $\mathcal{A}$) such that $KB \models C(a)$.

2.2 Rough Description Logics

DLs are suitable for modeling crisp knowledge, but they cannot easily be used to model approximate information. For example, no explicit mechanism is provided when a definition is not commonly agreed upon, or when exceptions need to be captured. Rough DLs (RDLs) attempt to close this gap in a conceptually simple way.

The basic idea behind RDLs is to approximate an uncertain concept C by giving an upper and a lower bound. The upper approximation of C, denoted $\overline{C}$, is the set of all individuals that possibly belong to C, while the lower approximation of C, denoted $\underline{C}$, is the set of all individuals that definitely belong to C. Traditionally, this is modeled using subsumption axioms; in pure DL modeling, the relation between C and its approximations $\underline{C}$ and $\overline{C}$ is $\underline{C} \sqsubseteq C \sqsubseteq \overline{C}$.

RDLs are not restricted to particular DLs, and can be defined for an arbitrary DL $\mathcal{DL}$. Its RDL language $\mathcal{RDL}$ has the lower and upper approximation as additional unary concept constructors, i.e., if C is a concept in $\mathcal{RDL}$, then also $\overline{C}$ and $\underline{C}$ are concepts in $\mathcal{RDL}$. The notions of *rough TBox* and *ABox*, as well as *rough knowledge base* then canonically extend their classical counterparts.

Example 2.1 (Advertising Campaign). Suppose that we want to use some pieces of data collected from the Web to find a group of people to serve as addressees for the advertising campaign of a new product. Clearly, the collected pieces of data are in general highly incomplete and uncertain. The DL concept *Addressee* may now be approximated from below by all the definite addressees and from above by all the potential addressees. So, we can use a DL to specify the TBox knowledge about *Addressee*, and in the same time specify the ABox knowledge about which people are definite and potential addressees, i.e., belong to the two concepts $\underline{\mathit{Addressee}}$ and $\overline{\mathit{Addressee}}$, respectively. ■

A *rough interpretation* is a triple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, R^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a domain of objects, $\cdot^{\mathcal{I}}$ is an interpretation function, and $R^{\mathcal{I}}$ is an equivalence (i.e., reflexive, symmetric, and transitive) relation over $\Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ maps RDL concepts to subsets of the domain $\Delta^{\mathcal{I}}$, and atomic roles to binary relations over $\Delta^{\mathcal{I}}$. It interprets the classical DL constructs and atomic concepts as usual, and the new constructs as follows:

- $(\overline{C})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$,
- $(\underline{C})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \forall b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$.

Intuitively, the upper approximation of a concept C covers the elements of a domain with the *typical properties* of C, while the lower approximation of C contains the *prototypical elements* of C.

Example 2.2 (Advertising Campaign cont'd). To define the definite and potential addressees for the advertising campaign of a new product, we may exploit a classification of people into equivalence classes. For example, people with an income above 1 million dollars may be definite addressees for the advertising campaign of a new Porsche, while people with an income above 100 000 dollars may be potential addressees, and people with an income below 10 000 dollars may not be addressees. ■

One of the advantages of this way of modeling uncertain concepts is that reasoning comes for free. Indeed, reasoning with approximations can be reduced to standard DL reasoning, by translating RDL concepts into classical DL concepts with a special reflexive and symmetric role.

A translation function for RDL concepts $\cdot^t : \mathcal{RDL} \mapsto \mathcal{DL}$ is defined as follows (introducing the new atomic role R for the indiscernibility relation): For every $\mathcal{RDL}$ concept C, the $\mathcal{DL}$ concept C^t is obtained from C by recursively (over the structure of C) replacing every $\overline{D}$ and $\underline{D}$ in C by $\exists R.D$ and $\forall R.D$, respectively, and using the identical mapping for all other constructs and atomic concepts. The translation function is naturally extended to axioms and knowledge bases (see [23]).

For any DL $\mathcal{DL}$ with universal and existential quantification, and reflexive, symmetric, and transitive roles, there is no increase in expressiveness, i.e., RDLs can be simulated in (almost) standard DLs: an $\mathcal{RDL}$ concept C is satisfiable in a rough interpretation relative to $\mathcal{T}$ iff the $\mathcal{DL}$ concept C^t is satisfiable relative to $\mathcal{T}^t$ [23]. In the presence of negation, other inference problems (such as subsumption checking) can be reduced to checking concept satisfiability (and finally to checking ABox satisfiability). Since the translation is linear, the complexity of reasoning in an RDL is the same as the one of reasoning in its DL counterpart with quantifiers as well as reflexive, symmetric, and transitive roles.

Since RDLs do not specify the nature of the indiscernibility relation, except prescribing its encoding as a (special) new equivalence relation, we introduce possible ways for defining it. The first one (see Section 3) makes it depend on a specific set of concepts determining the indiscernibility of the individuals relative to a specific context described by the concepts in the knowledge base. Then (see Section 4), we also define the indiscernibility relation in terms of a similarity measure (based on a context of features), which allows for relaxing the discernibility using a tolerance threshold. In case an indiscernibility relation cannot be specified (e.g., due to lack of knowledge), crisp descriptions of the concept approximations may be learned (see Section 5).

3 Contextual Indiscernibility Relations

In this section, we first define the notion of a context via a collection of DL concepts. We then introduce indiscernibility relations based on such contexts. We finally define upper and lower approximations of DL concepts using these notions, and we provide some theoretical results about them.

It is well known that classification by analogy cannot be really general-purpose, since the number of features on which the analogy is made may be very large [21]. The key point is that indiscernibility is not absolute, but, rather, an induced notion, which depends on the specific contexts of interest. Instead of modeling indiscernibility through a single relation in the interpretation, one may consider diverse contexts, each giving rise to a different relation, which determines also different ways of approximating uncertain concepts. We first recall the notion of projection function [6].

Definition 3.1 (projection). Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation, and let F be a DL concept. Then, the projection function $\pi_F^{\mathcal{I}} : \Delta^{\mathcal{I}} \mapsto \{0, 1\}$ is defined as follows:

$$\forall a \in \Delta^{\mathcal{I}} : \quad \pi_F^{\mathcal{I}}(a) = \begin{cases} 1 & a \in F^{\mathcal{I}}; \\ 0 & \text{otherwise.} \end{cases}$$

We define a *context* as a finite set of relevant features in the form of DL concepts, which may encode context information for the similarity to be measured [12].

Definition 3.2 (context). A *context* is a set of DL concepts $\mathsf{C} = \{F_1, \ldots, F_m\}$.

Example 3.1 (Advertising Campaign cont'd). One possible context C for the advertising campaign of a new product is given as follows:

$$\mathsf{C} = \{\mathit{SalaryAboveMillion}, \mathit{HouseOwner}, \mathit{Manager}\},$$

where *SalaryAboveMillion*, *HouseOwner*, and *Manager* are DL concepts. ■

Two individuals a and b are *indiscernible* relative to the context $\mathsf{C} = \{F_1, \ldots, F_m\}$ iff $\pi_{F_i}(a) = \pi_{F_i}(b)$ for all $i \in \{1, \ldots, m\}$. This induces an equivalence relation. Note that one may define multiple such relations by considering different contexts.

Definition 3.3 (indiscernibility relation). Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation, and let $\mathsf{C} = \{F_1, \ldots, F_m\}$ be a context. Then, the indiscernibility relation $R_{\mathsf{C}}^{\mathcal{I}}$ induced by C under $\mathcal{I}$ is defined as follows:

$$R_{\mathsf{C}}^{\mathcal{I}} = \left\{(a, b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}) \mid \forall i \in \{1, \ldots, m\} : \pi_{F_i}^{\mathcal{I}}(a) = \pi_{F_i}^{\mathcal{I}}(b)\right\}.$$

Any indiscernibility relation splits $\Delta^{\mathcal{I}}$ in a partition of equivalence classes (also known as *elementary sets*) denoted $[a]_{\mathsf{C}}$, for a generic individual a. Each class naturally induces a concept, denoted C_a.

Example 3.2 (Advertising Campaign cont'd). Consider again the context C of Example 3.1. Observe that C defines an indiscernibility relation on the set of all people, which is given by the extensions of all atomic concepts constructed from C as its equivalence classes. For example, one such atomic concept is the conjunction of *SalaryAboveMillion*, *HouseOwner*, and *Manager*; another one is the conjunction of *SalaryAboveMillion*, *HouseOwner*, and $\neg$ *Manager*. ■

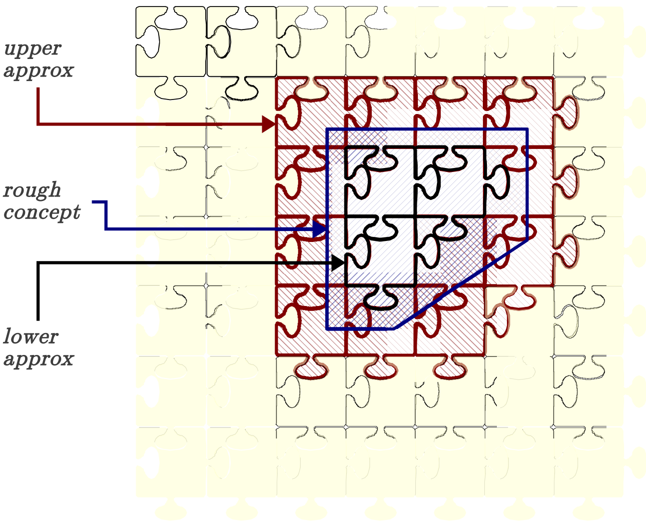

Fig. 1. Lower and upper approximations of rough concepts

Thus, a C-*definable* concept has an extension that corresponds to the union of elementary sets. The other concepts may be approximated as usual (we give a slightly different definition of the approximations relative to those in Section 2.2).

Definition 3.4 (contextual approximations). Let $\mathsf{C} = \{F_1, \ldots, F_m\}$ be a context, let D be a DL concept, and let $\mathcal{I}$ be an interpretation. Then, the *contextual upper and lower approximations* of D relative to C, denoted $\overline{D}^{\mathsf{C}}$ and $\underline{D}_{\mathsf{C}}$, respectively, are defined as follows:

- $(\overline{D}^{\mathsf{C}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \mathcal{I} \not\models \mathsf{C}_a \sqcap D \sqsubseteq \bot\}$,
- $(\underline{D}_{\mathsf{C}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \mathcal{I} \models \mathsf{C}_a \sqsubseteq D\}$.

Figure 1 illustrates the contextual upper and lower approximations. The partition is determined by the feature concepts included in the context, each block standing for one of the C-definable concepts. The blocks inscribed in the concept polygon represent its lower approximation, while the blocks having a nonempty intersection with the concept polygon stand for its upper approximation.

These approximations can be encoded in a DL knowledge base through special indiscernibility relationships, as in [23], so to exploit standard reasoners for implementing inference services (with crisp answers). Alternatively, new constructors for contextual rough approximation may be defined to be added to the standard ones in the specific DL.

Following an analogous approach to the one presented in [18], it is easy to see that the following properties hold for these operators:

Proposition 3.1 (properties). *Let* $\mathsf{C} = \{F_1, \ldots, F_m\}$ *be a context, let* D *and* E *be two DL concepts. Then:*

1. $\underline{\bot}_{\mathsf{C}} = \overline{\bot}^{\mathsf{C}} = \bot$,
2. $\underline{\top}_{\mathsf{C}} = \overline{\top}^{\mathsf{C}} = \top$,
3. $\underline{D \sqcup E}_{\mathsf{C}} \sqsupseteq \underline{D}_{\mathsf{C}} \sqcup \underline{E}_{\mathsf{C}}$,
4. $\overline{D \sqcup E}^{\mathsf{C}} = \overline{D}^{\mathsf{C}} \sqcup \overline{E}^{\mathsf{C}}$,
5. $\underline{D \sqcap E}_{\mathsf{C}} = \underline{D}_{\mathsf{C}} \sqcap \underline{E}_{\mathsf{C}}$,
6. $\overline{D \sqcap E}^{\mathsf{C}} \sqsubseteq \overline{D}^{\mathsf{C}} \sqcap \overline{E}^{\mathsf{C}}$,
7. $\underline{\neg D}_{\mathsf{C}} = \neg \overline{D}^{\mathsf{C}}$,
8. $\overline{\neg D}^{\mathsf{C}} = \neg \underline{D}_{\mathsf{C}}$,
9. $\underline{\underline{D}_{\mathsf{C}}}_{\mathsf{C}} = \underline{D}_{\mathsf{C}}$,
10. $\overline{\overline{D}^{\mathsf{C}}}^{\mathsf{C}} = \overline{D}^{\mathsf{C}}$.

4 Numerical Extensions

In this section, the indiscernibility relation is expressed in terms of a similarity measure. We introduce contextual similarity measures, and we discuss the aspect of finding optimal contexts. We finally describe how indiscernibility relations can be defined on top of tolerance functions.

4.1 Contextual Similarity Measures

Since indiscernibility can be graded in terms of the similarity between individuals, we propose a set of similarity functions, based on ideas that inspired a family of inductive distance measures [6, 2]:

Definition 4.1 (family of similarity functions). Let $KB = (\mathcal{T}, \mathcal{A})$ be a knowledge base. Given a context $\mathsf{C} = \{F_1, F_2, \ldots, F_m\}$, a family of similarity functions

$$s_p^{\mathsf{C}} : \mathsf{Ind}(\mathcal{A}) \times \mathsf{Ind}(\mathcal{A}) \mapsto [0, 1]$$

is defined as follows ($\forall a, b \in \mathsf{Ind}(\mathcal{A})$):

$$s_p^{\mathsf{C}}(a, b) := \frac{1}{m} \left[\sum_{i=1}^{m} \sigma_i(a, b)^p \right]^{\frac{1}{p}}, \tag{1}$$

where $p > 0$, and the *basic similarity function* σ_i ($\forall i \in \{1, \ldots, m\}$) is defined by:

$$\sigma_i(a, b) = \begin{cases} 1 & (KB \models F_i(a) \wedge KB \models F_i(b)) \vee (KB \models \neg F_i(a) \wedge KB \models \neg F_i(b)); \\ 0 & (KB \models \neg F_i(a) \wedge KB \models F_i(b)) \vee (KB \models F_i(a) \wedge KB \models \neg F_i(b)); \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

The rationale for these functions is that similarity between individuals is determined relative to a given context [12]. Two individuals are maximally similar relative to a given concept F_i if they exhibit the same behavior, i.e., both are instances of the concept or of its negation. Conversely, the minimal similarity holds when they belong to opposite concepts. By the open-world semantics, sometimes a reasoner cannot assess the concept-membership, hence, since both possibilities are open, an intermediate value is assigned to reflect such uncertainty.

As mentioned, instance-checking is used for assessing the value of the basic similarity functions. As this is known to be computationally expensive (also depending on the specific DL language), a simple look-up may be sufficient, especially for ontologies that are rich of explicit class-membership information (assertions). Hence, alternatively, for densely populated knowledge bases, the σ_i's can be efficiently approximated by defining them as follows ($\forall a, b \in \mathsf{Ind}(\mathcal{A})$):

$$\sigma_i(a,b) = \begin{cases} 1 & (F_i(a) \in \mathcal{A} \wedge F_i(b) \in \mathcal{A}) \vee (\neg F_i(a) \in \mathcal{A} \wedge \neg F_i(b) \in \mathcal{A}); \\ 0 & (F_i(a) \in \mathcal{A} \wedge \neg F_i(b) \in \mathcal{A}) \vee (\neg F_i(a) \in \mathcal{A} \wedge F_i(b) \in \mathcal{A}); \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

The parameter p in (1) was borrowed from the form of Minkowski's measures [24]. Once the context is fixed, the possible values for the similarity function are determined; hence, p has an impact on the granularity of the measure.

Furthermore, the uniform choice of the weights assigned to the similarity related to the various features in the sum $(1/m^p)$ may be replaced by assigning different weights reflecting the importance of a certain feature in discerning the various instances. A good choice may be based on the amount of *entropy* related to each feature concept (then the weight vector has only to be normalized) [2].

4.2 Optimization of the Contexts

It is worthwhile to note that Definition 4.1 introduces a family of functions that are parameterized on the choice of features.

Preliminarily, the very set of both atomic and defined concepts found in the knowledge base can be used as a context.[1] But the choice of the concepts to be included in the context C is crucial, both for the effectiveness of the measure and for the computational efficiency itself. Specifically, the required computational effort grows with the size of the context C.

As performed for inducing the pseudo-metric that inspired the definition of the similarity function [6], a preliminary phase may concern finding optimal contexts. This may be carried out by means of randomized optimization procedures.

Since the underlying idea in the definition of the functions is that similar individuals should exhibit the same behavior relative to the concepts in C, the context C should represent a sufficient number of (possibly redundant) features that are able to discriminate different individuals.

[1] Preliminary experiments, reported in [2], demonstrated the effectiveness of the similarity function using the very set of both atomic and defined concepts found in the knowledge base.

The problem may be regarded as a learning problem having as a goal finding an optimal context (given the knowledge base) provided that two crucial factors are considered/optimized:

- the *number* of concepts of the context C,
- the discriminating power of the concepts in C in terms of a *discernibility factor*, i.e., a measure of the amount of difference between individuals.

The learned discriminating concepts in C may be complex concepts that are built via the specific constructors of the underlying DL.

A possible solution to the learning problem has been discussed in [6], where a randomized optimization procedure is proposed. This solution is particularly well-suited when knowledge bases with large sets of individuals are considered.

4.3 Approximation by Tolerance

In [4], a less strict type of approximation is introduced, based on the notion of *tolerance*. Exploiting the similarity functions that have been defined in Section 4.1, it is easy to extend this kind of (contextual) approximation to the case of RDLs.

Let a *tolerance function* on a set U be any function $\tau : U \times U \mapsto [0, 1]$ such that for all $a, b \in U$, it holds that $\tau(a, a) = 1$ and $\tau(a, b) = \tau(b, a)$.

Considering a tolerance function τ on a (*universal*) set U and a *tolerance threshold* $\theta \in [0, 1]$, a *neighborhood function* $\nu : U \mapsto 2^U$ is defined as follows:

$$\nu_\theta(a) = \{b \in U \mid \tau(a, b) \geq \theta\}.$$

For each element $a \in U$, the set $\nu_\theta(a)$ is the *neighborhood* of a.

Consider now the domain $\Delta^\mathcal{I}$ of an interpretation $\mathcal{I}$ as a universal set, a similarity function s_p^C on $\Delta^\mathcal{I}$ (for some context C) as a tolerance function, and a threshold $\theta \in [0, 1]$. It is then easy to derive a *tolerance relation*[2], i.e., a reflexive and symmetric relation on $\Delta^\mathcal{I}$, inducing tolerance classes that consist of individuals within a certain degree of similarity, indicated by the threshold: $[a]_{\mathsf{C},\theta} = \nu_\theta(a)$. The notions of upper and lower approximation relative to the tolerance relation induced by C and θ descend straightforwardly:

- $(\overline{D})^\mathcal{I} = \{a \in \Delta^\mathcal{I} \mid \exists b \in \Delta^\mathcal{I} : s_p^\mathsf{C}(a, b) \geq \theta \wedge b \in D^\mathcal{I}\}$,
- $(\underline{D})^\mathcal{I} = \{a \in \Delta^\mathcal{I} \mid \forall b \in \Delta^\mathcal{I} : s_p^\mathsf{C}(a, b) \geq \theta \rightarrow b \in D^\mathcal{I}\}$.

Given the similarity measure defined in Section 4.1 as a tolerance function, the approximation by tolerance allows a less strict approximation with respect to the adoption of the indiscernibility relation exploited for the case of the contextual approximation (see Section 3). The granularity of the approximation is specifically controlled by the threshold. Indeed, if θ is (very close to) 1, then we obtain almost the indiscernibility relation for the contextual approximation. Considering lower values for θ, additional individuals will be included in the neighborhood $\nu_\theta(a)$ of a given individual a. This aspect may result to be particularly useful when not enough information is available for defining a suitable context of interest C.

[2] Transitivity is not necessary, however, the case of an indiscernibility relation can be considered with the equivalence classes $[a]_\theta = \bigcap\{\nu_\theta(b) \mid a \in \nu_\theta(b)\}$.

Example 4.1 (Advertising Campaign cont'd). Given the context

$$\mathsf{C} = \{\mathit{SalaryAboveMillion}, \mathit{HouseOwner}, \mathit{Manager}\}$$

introduced in Example 3.1, $\theta = 0.8$, and the similarity function s_p^{C} defined in Section 4.1, the concepts $\underline{\mathit{Addressee}}$ and $\overline{\mathit{Addressee}}$ will be given by

- $(\overline{\mathit{Addressee}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists b \in \Delta^{\mathcal{I}} : s_p^{\mathsf{C}}(a,b) \geq 0.8 \wedge b \in \mathit{Addressee}^{\mathcal{I}}\}$,
- $(\underline{\mathit{Addressee}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \forall b \in \Delta^{\mathcal{I}} : s_p^{\mathsf{C}}(a,b) \geq 0.8 \rightarrow b \in \mathit{Addressee}^{\mathcal{I}}\}$. ■

Note that these approximations depend on the threshold. Thus, we have a numerical way to control the degree of indiscernibility that is needed to model uncertain concepts. This applies both to the standard RDL setting and to the new contextual one presented in the previous section.

Alternatively, by learning an intensional concept description (see the next Section 5) for the neighborhood $\nu_\theta(a)$, lower and upper approximations of a given concept D may be defined as for the case of the contextual approximation (see Section 3).

5 Learning Crisp Definitions of Rough Concepts

There may be cases where modeling uncertain knowledge is a difficult task, even in the framework of RDLs. One main problem is that a domain expert may not always have a clear idea about the concepts to model via lower and upper concept approximations, thus having incomplete besides rough concept definitions. Another problem are the difficulties in defining a suitable indiscernibility relation. Even the indiscernibility relations in Sections 3 and 4 are based on the notion of a context, which for some cases may be difficult to define (see, e.g., the *Sepsis* example in [23]). Furthermore, even the translation function (presented at the end of Section 2.2) for transforming RDL concepts into crisp concepts, via an appropriate predicate for the indiscernibility relation, may be hard to apply in practice in DLs with low expressiveness. Furthermore, a domain expert often has a clear idea about counterexamples to a concept definition, but may not be able to give an explanation or a clear definition for them.

To cope with these problems, and to still be able to represent uncertain knowledge in RDLs, we propose an alternative way, grounded on DL concept learning methods, for describing lower and upper approximations of a given concept.

Since a domain expert often has a clear idea about counterexamples to a given concept definition, we assume that he/she is able to supply (a) a set of positive examples for the upper (resp., lower) approximation of a given concept C, i.e., a set of individuals standing as the possible (resp., certain) instances of the concept C, and (b) a set of negative examples for the upper (resp., lower) approximation of C, i.e., a set of individuals that are surely not instances of the upper (resp., lower) approximation of C.

Given these sets of examples, the problem now is to find a suitable crisp definition for them. Specifically, the problem can be formally defined as follows:

Definition 5.1 (learning problem). Let $KB = (\mathcal{T}, \mathcal{A})$ be a knowledge base. Then, given

– $\mathsf{Ind}(\mathcal{A})$ the set of all individuals occurring in $\mathcal{A}$,
– a set of positive and negative examples $\mathsf{Ind}^+_C(\mathcal{A}) \cup \mathsf{Ind}^-_C(\mathcal{A}) \subseteq \mathsf{Ind}(\mathcal{A})$ for the upper (resp., lower) approximation of a given concept C,

we build a concept definition $\widehat{C}$ such that

$$KB \models \widehat{C}(a) \quad \forall a \in \mathsf{Ind}^+_C(\mathcal{A}) \qquad \text{and} \qquad KB \models \neg\widehat{C}(b) \quad \forall b \in \mathsf{Ind}^-_C(\mathcal{A}).$$

Example 5.1 (Advertising Campaign cont'd). Consider again Examples 2.1 and 2.2, where the RDL concepts $\underline{Addressee}$ and $\overline{Addressee}$ are introduced, representing the definite and potential addressees, respectively, for an advertising campaign for a new product (i.e., Porsche). Now, a crisp DL definition for each of the two concepts has to be given. Suppose now that no indiscernibility function is adopted and/or specified, because the domain expert does not have enough knowledge (e.g., for defining a suitable context C), but the domain expert is able to identify some instances (i.e., individuals of the knowledge base, i.e., instances of some concepts in the knowledge base) that are definitely addressees (positive examples) and that are surely not addressees (negative examples). This information is exploited for learning an intensional concept description such that all positive examples are instances of the learned concept and that all negative examples are instances of the negation of the learned concept. The same process can be applied for the concept $\overline{Addressee}$. In this way, a crisp description for an RDL concept can be given without adopting any indiscernibility function. ■

The definition given above can be interpreted as a generic supervised concept learning task. The problem consists of finding a DL concept definition $\widehat{C}$ such that all positive examples are instances of $\widehat{C}$, while all negative examples are instances of $\neg\widehat{C}$. This problem is well-studied in the literature, resulting in different inductive learning methods that are grounded on the (greedy) exploration of the search space by the adoption of suitable refinement operators for DL representations [14, 15]. Among the most well-known algorithms and systems, there are DL-FOIL [7], DL-LEARNER[3] [16], and TERMITIS [11]. Hence, given the set of the positive and negative examples for the upper (resp., lower) approximation of a concept C, the crisp definitions of the approximate concepts can be learned by adopting one of the systems cited above.

Alternatively/additionally, we may also be interested in assessing/learning the crisp description of the upper (resp., lower) approximation of a crisp concept D that is already existing in the knowledge base. In this case, finding a domain expert who provides the set of positive and negative examples for the upper (resp., lower) approximation of D may not always be possible. The problem of learning a crisp concept description for the upper (resp., lower) approximation of D is now shifted to the problem of determining the positive and negative examples for the upper (resp., lower) approximation of D. In the following, the possible solutions are illustrated.

Definition 5.2 (positive/negative examples for lower approximation). Let $KB = (\mathcal{T}, \mathcal{A})$ be a knowledge base. Then, given

– $\mathsf{Ind}(\mathcal{A})$ the set of all individuals occurring in $\mathcal{A}$,
– a target atomic concept D,

[3] `http://dl-learner.org/Projects/DLLearner`

we define

– the set of positive examples as $\mathsf{Ind}^{+}_{\underline{D}}(\mathcal{A}) = \{a \in \mathsf{Ind}(\mathcal{A}) \mid KB \models D(a)\}$,
– the set of negative examples as $\mathsf{Ind}^{-}_{\underline{D}}(\mathcal{A}) = \{a \in \mathsf{Ind}(\mathcal{A}) \mid KB \not\models D(a)\}$.

The set of the positive examples for the lower approximation of the concept D is given by all individuals of the knowledge base that are instances[4] of D, while the set of negative examples is given by all individuals for which it is not possible to prove that they are instances of D. This set includes both the individuals that are instances of $\neg D$ and the individuals for which the reasoner is not able to give any reply due to the OWA.

Definition 5.3 (positive/negative examples for upper approximation). Let $KB = (\mathcal{T}, \mathcal{A})$ be a knowledge base. Then, given

– $\mathsf{Ind}(\mathcal{A})$ the set of all individuals occurring in $\mathcal{A}$,
– a target atomic concept D,

we define

– the set of positive examples as $\mathsf{Ind}^{+}_{\overline{D}}(\mathcal{A}) = \{a \in \mathsf{Ind}(\mathcal{A}) \mid KB \not\models \neg D(a)\}$,
– the set of negative examples as $\mathsf{Ind}^{-}_{\overline{D}}(\mathcal{A}) = \{a \in \mathsf{Ind}(\mathcal{A}) \mid KB \models \neg D(a)\}$.

The set of the negative examples for the upper approximation of the concept D is given by all individuals of the knowledge base that are instances of $\neg D$, while the set of positive examples is given by all individuals for which it is not possible to prove that they are instances of $\neg D$ (e.g., because of the absence of disjointness axioms in the considered ontology). This set includes both the individuals that are instances of D and the individuals for which the reasoner is not able to give any reply due to the OWA.

Once the set of positive and negative examples for the lower (resp., upper) approximation of D have been determined, the crisp definition for the lower (resp., upper) approximation of D can be learned as illustrated above. Note, however, that the learned definitions may be noisy when a high percentage of unlabeled examples (due to the OWA) is included in the set of negative (resp., positive) examples. To cope with this problem, alternative learning methods such as methods for learning from positive (and unlabeled) examples [3, 25] only were investigated and can be exploited.

6 Summary and Outlook

Inspired by previous works on dissimilarity measures in DLs, we have defined a notion of context, which allows to extend the indiscernibility relation adopted by rough DLs, thus allowing for various kinds of approximations of uncertain concepts within the same knowledge base. It also saves the advantage of encoding the relation in the same DL language, thus allowing for reasoning with uncertain concepts through standard tools, obtaining crisp answers to queries.

Alternatively, these approximations can be implemented as new modal-like language operators. Some properties of the approximations deriving from the theory of rough sets have also been investigated.

[4] Here, *concept retrieval* may be adopted.

A novel family of semantic similarity functions for individuals has also been defined based on their behavior relative to a number of features (concepts). The functions are language-independent, being based on instance-checking (or ABox look-up). This allows for defining further kinds of graded approximations based on the notion of tolerance relative to a certain threshold.

Since data can be classified into indiscernible clusters, unsupervised learning methods for grouping individuals on the grounds of their similarity can be used for the definition of an equivalence relation [13, 6, 8]. Besides, it is also possible to learn rough DL concepts from the explicit definitions of the instances of particular concepts [14, 15, 7].

Acknowledgments. This work was partially supported by the Engineering and Physical Sciences Research Council (EPSRC) grant EP/J008346/1 (PrOQAW), the European Research Council under the EU's 7th Framework Programme (FP7/2007-2013/ERC) grant 246858 (DIADEM), a Google Research Award, and a Yahoo! Research Fellowship. Many thanks also to the reviewers of this paper and its URSW-2008 abstract for their useful and constructive comments, which have helped to improve this work.

References

[1] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook. Cambridge University Press (2003)

[2] d'Amato, C., Fanizzi, N., Esposito, F.: Query Answering and Ontology Population: An Inductive Approach. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 288–302. Springer, Heidelberg (2008)

[3] De Comité, F., Denis, F., Gilleron, R., Letouzey, F.: Positive and Unlabeled Examples Help Learning. In: Watanabe, O., Yokomori, T. (eds.) ALT 1999. LNCS (LNAI), vol. 1720, pp. 219–230. Springer, Heidelberg (1999)

[4] Doherty, P., Grabowski, M., Łukaszewicz, W., Szalas, A.: Towards a framework for approximate ontologies. Fundamenta Informaticae 57(2-4), 147–165 (2003)

[5] Donini, F., Lenzerini, M., Nardi, D., Nutt, W.: An epistemic operator for description logics. Artificial Intelligence 100(1/2), 225–274 (1998)

[6] Fanizzi, N., d'Amato, C., Esposito, F.: Randomized metric induction and evolutionary conceptual clustering for semantic knowledge bases. In: Proceedings of the 16th Conference on Information and Knowledge Management, CIKM 2007, pp. 51–60. ACM Press (2007)

[7] Fanizzi, N., d'Amato, C., Esposito, F.: DL-FOIL Concept Learning in Description Logics. In: Železný, F., Lavrač, N. (eds.) ILP 2008. LNCS (LNAI), vol. 5194, pp. 107–121. Springer, Heidelberg (2008)

[8] Fanizzi, N., d'Amato, C., Esposito, F.: Conceptual Clustering and Its Application to Concept Drift and Novelty Detection. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 318–332. Springer, Heidelberg (2008)

[9] Fanizzi, N., d'Amato, C., Esposito, F.: Statistical Learning for Inductive Query Answering on OWL Ontologies. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 195–212. Springer, Heidelberg (2008)

[10] Fanizzi, N., d'Amato, C., Esposito, F., Lukasiewicz, T.: Representing uncertain concepts in rough description logics via contextual indiscernibility relations. In: Proceedings of the 4th International Workshop on Uncertainty Reasoning for the Semantic Web, URSW 2008. CEUR Workshop Proceedings, vol. 423. CEUR-WS.org (2008)

[11] Fanizzi, N., d'Amato, C., Esposito, F.: Induction of Concepts in Web Ontologies through Terminological Decision Trees. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010, Part I. LNCS, vol. 6321, pp. 442–457. Springer, Heidelberg (2010)
[12] Goldstone, R., Medin, D., Halberstadt, J.: Similarity in context. Memory and Cognition 25(3), 237–255 (1997)
[13] Hirano, S., Tsumoto, S.: An indiscernibility-based clustering method. In: Proceedings of the 2005 IEEE International Conference on Granular Computing, pp. 468–473. IEEE Computer Society (2005)
[14] Iannone, L., Palmisano, I., Fanizzi, N.: An algorithm based on counterfactuals for concept learning in the Semantic Web. Applied Intelligence 26(2), 139–159 (2007)
[15] Lehmann, J., Hitzler, P.: A Refinement Operator Based Learning Algorithm for the $\mathcal{ALC}$ Description Logic. In: Blockeel, H., Ramon, J., Shavlik, J., Tadepalli, P. (eds.) ILP 2007. LNCS (LNAI), vol. 4894, pp. 147–160. Springer, Heidelberg (2008)
[16] Lehmann, J.: DL-Learner: Learning concepts in description logics. Journal of Machine Learning Research 10, 2639–2642 (2009)
[17] Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the Semantic Web. Journal of Web Semantics 6(4), 291–308 (2008)
[18] Jiang, Y., Wang, J., Tang, S., Xiao, B.: Reasoning with rough description logics: An approximate concepts approach. Information Sciences 179(5), 600–612 (2009)
[19] Keet, C.M.: On the feasibility of description logic knowledge bases with rough concepts and vague instances. In: Proceedings of the 23rd International Workshop on Description Logics, DL 2010. CEUR Workshop Proceedings, vol. 573. CEUR-WS.org (2010)
[20] Keet, C.M.: Ontology Engineering with Rough Concepts and Instances. In: Cimiano, P., Pinto, H.S. (eds.) EKAW 2010. LNCS, vol. 6317, pp. 503–513. Springer, Heidelberg (2010)
[21] Mitchell, T.: Machine Learning. McGraw-Hill (1997)
[22] Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning About Data. Kluwer Academic Publishers (1991)
[23] Schlobach, S., Klein, M.C.A., Peelen, L.: Description logics with approximate definitions — precise modeling of vague concepts. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI 2007, pp. 557–562 (2007)
[24] Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search — The Metric Space Approach. Advances in Database Systems. Springer (2007)
[25] Zhang, B., Zuo, W.: Learning from positive and unlabeled examples: A survey. In: Proceeding of the International Symposium on Information Processing, ISP 2008, pp. 650–654 (2008)