A Dissimilarity Measure for the $\mathcal{ALC}$ Description Logic

Claudia d'Amato, Nicola Fanizzi, Floriana Esposito

Dipartimento di Informatica, Università degli Studi di Bari
Campus Universitario, Via Orabona 4, 70125 Bari, Italy
{claudia.damato, fanizzi, esposito}@di.uniba.it

Abstract. This work presents a dissimilarity measure for an expressive Description Logic endowed with the principal constructors employed in the standard representations for ontological knowledge. In particular, the focus is on the definition of a dissimilarity measure for the $\mathcal{ALC}$ description logic based both on the syntax and on the semantics of the descriptions. The measure is shown to be applicable to assess the dissimilarity for cases involving individuals.

1 Introduction

Ontological knowledge plays a key role for interoperability in the Semantic Web perspective. Nowadays, standard ontology markup languages are supported by well-founded semantics of Description Logics (DLs) together with a series of available automated reasoning services [BCM+03]. However, several tasks in an ontology life-cycle [SS04], such as their construction and/or integration are still almost entirely delegated to knowledge engineers.

Similarity measures play an important role in information retrieval and information integration. In the Semantic Web perspective, the construction of the knowledge bases should be supported by automated inductive inference services. The induction of structural knowledge is not new in machine learning, especially in the context of *concept formation*, where clusters of similar objects are aggregated in hierarchies according to heuristic criteria or similarity measures. Almost all of these methods apply to zero-order representations while, as mentioned above, ontologies are expressed by means of fragments of first-order logic. Yet, the problem of the induction of structural knowledge turns out to be hard in first-order logic or equivalent representations. In *relational learning* attempts have been made to extend the standard ILP techniques towards hybrid representations based on both clausal and description logics [RV00, Kie02]. In order to cope with the problem complexity, these methods are based on a heuristic search and generally implement algorithms that tend to induce overly specific concept definitions which may suffer for poor predictive capabilities, (such as the LCS operator [CH94]).

In this perspective, we introduce a novel dissimilarity measure between concept descriptions based also on semantics, which is suitable for expressive DLs

like $\mathcal{ALC}$ [SSS91, BCM+03]. Since a merely syntactic approach has proven too weak to enforce standard inferences (namely subsumption) [BCM+03], when expressive DLs are taken into account, a different approach is necessary. Also a dissimilarity measure, then, should be founded on the underlying semantics, rather than on the syntactic structure of concept descriptions.

Besides measuring the dissimilarity of two concept descriptions, we also propose a individual-to-concept and an individual-to-individual dissimilarity, based on notion of *most specific concept* of an individual (see [BCM+03], chap. 2) for applying the same measure to these different cases.

Similarity measures may support also retrieval and ontology integration. Such a measure can be the basis for adapting an existing clustering method to this representation (or devising a new one) operating in a top-down (*partitional*) or bottom-up (*agglomerative*) fashion. Moreover, the dissimilarity measure can be also employed for *Information Retrieval* or *Information Integration* purposes applied to DL knowledge bases and also for *Case-based Reasoning* systems.

As discussed in the following, the method can effectively compute the dissimilarity measure with a complexity which depends on the complexity of standard inferences as a baseline.

The remainder of the paper is organized as follows. The next section reviews related work on related measures. In Sect. 3 the representation language is presented. The dissimilarity measure is illustrated in Sect. 4 and is discussed in Sect. 5. Possible developments of the method are examined in Sect. 6.

2 Related Work

Similarity and dissimilarity measures play an important role in information retrieval and information integration. Recent investigations in these fields have emphasized the use of ontologies and semantic similarity functions as a mechanism for comparing concepts and/or concept instances that can be retrieved or integrated across heterogeneous repositories [JC97, GMV99].

A semantic similarity is typically determined as a function of the *path distance* between terms in the hierarchical structure underlying the ontology [BHP94]. Other methods to assess semantic similarity within a single ontology are *feature matching* [Tve97] and *information content* [Res99]. The former approach uses both common and discriminant features among concepts and/or concept instances in order to compute the semantic similarity. The latter methods are founded on *Information Theory*. They define a similarity measure between two concepts within a concept hierarchy in terms of the amount of information conveyed by the immediate super-concept subsuming the concepts under comparison. This is a measure of the variation of information crossing from a description level to a more general one.

A recent work [WB99] presents a number of measures for comparing concepts located in different and possibly heterogeneous ontologies. The following requirements are made for this measure:

- the formal representation supports inferences such as *subsumption*;

– local concepts in different ontologies inherit their definitional structure from concepts in a shared ontology.

This study assumes that the intersection of sets of concept instances is an indication of the correspondence between these concepts. Three main types of measures for comparing concept descriptions are discussed in this work:

1. *filter* measures based on a path-distance
2. *matching* measures based on graph matching establish one-to-one correspondence between elements of the concept descriptions, and
3. *probabilistic* measures that give the correspondence in terms of the joint distribution of concepts.

Other similarity measures have been developed to compute similarity values among classes (concepts) belonging to different ontologies. These measures are able to take into account the difference in the levels of explicitness and formalization of the different ontology specifications. Particularly, in [RE03] a similarity function determines similar entity classes by using a matching process making use of synonym sets, semantic neighborhood, and discriminating features that are classified into parts, functions, and attributes.

Another approach [Man00], aimed at finding commonalities among concepts or among assertions, employs the *Least Specific Concept* operator (LCS [CH94, BCM$^+$03]) that computes the most specific generalization of the input concepts (with respect to subsumption, see the next section for a formal definition). This approach is generally intended for information retrieval purposes. Considered a knowledge base and a query concept, a filter mechanism selects another concept from the knowledge base that is relevant for the query concept. Then the LCS of the two concepts is computed and finally all concepts subsumed by the LCS are returned.

Most of the cited works adopt a semantic approach in conjunction with the structure of the considered concept descriptions. Thus, they are liable to the phenomenon of the rapid growth of the description granularity. Besides the syntactic structure of concept descriptions becomes much less important when richer DL representations are adopted due to the expressive operators that can be employed. For these reasons, we have decided to focus our attention to a measure which is strongly based on semantics. In this respect, to the best of our knowledge, there has been no comparable effort in the literature, except the ideas in [BWH05].

3 The Representation Language

In relational learning, several solutions have been proposed for the adoption of an expressive fragment of first-order logic endowed with efficient inference procedures. Alternatively, the data model of a knowledge base can be expressed by means of DL concept languages which are empowered with precise semantics and effective inference services [BCM$^+$03]. Besides, most of the ontology markup

Table 1. $\mathcal{ALC}$ constructors and their meaning.

Name	Syntax	Semantics
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
concept	C	$C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
concept negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
concept conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
concept disjunction	$C_1 \sqcup C_2$	$C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$
existential restriction	$\exists R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}((x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}})\}$
universal restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}}((x,y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}})\}$

languages for the Semantic Web (e.g., OWL) are founded in Description Logics: representation languages borrow and implement the typical constructors of the DL languages. Although it can be assumed that annotations and conceptual models are maintained and transported using the XML-based languages mentioned above, the syntax of the representation adopted here is taken from the standard constructors proposed in the DL literature [BCM$^+$03]. These DL representations turn out to be both sufficiently expressive and efficient from an inferential viewpoint.

In this section we recall syntax and semantics for the reference representation $\mathcal{ALC}$ [SSS91] which is adopted in the rest paper for it turns out to be sufficiently expressive to support most of the principal constructors of an ontology markup language for the Semantic Web.

In a DL language, primitive *concepts*, denoted with names taken from $N_C = \{C, D, \ldots\}$, are interpreted as subsets of a certain domain of objects (resources) or equivalently as unary relation on such domain and primitive *roles*, denoted with names taken from $N_R = \{R, S, \ldots\}$, are interpreted as binary relations on such a domain (properties). Complex concept descriptions can be built using primitive concepts and roles by means of the constructors in Table 1. Their semantics is defined by an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the *domain* of the interpretation and the functor $\cdot^{\mathcal{I}}$ stands for the *interpretation function*, mapping the intension of concepts and roles to their extension.

A *knowledge base* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains two components: A *T-box* $\mathcal{T}$ and an *A-box* $\mathcal{A}$. $\mathcal{T}$ is a set of concept definitions $C \equiv D$, meaning $C^{\mathcal{I}} = D^{\mathcal{I}}$, where C is the concept name and D is a description given in terms of the language constructors. $\mathcal{A}$ contains extensional assertions on concepts and roles, e.g. $C(a)$ and $R(a, b)$, meaning, respectively, that $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$; $C(a)$ and $R(a, b)$ are said respectively instance of the concept C and instance of the role R, more generally it is said (without loss of generality) that the individual a is instance of the concept C and the same for the role. A notion of *subsumption* between concepts is given in terms of the interpretations:

Definition 3.1 (subsumption). *Given two concept descriptions C and D, C* subsumes *D, denoted by $C \sqsupseteq D$, iff for every interpretation $\mathcal{I}$ it holds that $C^{\mathcal{I}} \supseteq D^{\mathcal{I}}$.*

Axioms based on subsumption ($C \sqsupseteq D$) are generally also allowed in the T-boxes as partial definitions. Indeed, $C \equiv D$ amounts to $C \sqsupseteq D$ and $D \sqsupseteq C$.

Example 3.1. An instance of concept definition in the proposed language is:

$$\mathsf{Father} \equiv \mathsf{Male} \sqcap \exists \mathsf{hasChild}.\mathsf{Person}$$

which corresponds to the sentence: *"a father is a male (person) that has some persons as his children".*

The following are instances of simple assertions:

$\mathsf{Male}(\mathsf{Leonardo})$, $\mathsf{Male}(\mathsf{Vito})$, $\mathsf{hasChild}(\mathsf{Leonardo}, \mathsf{Vito})$.

Supposing that $\mathsf{Male} \sqsubseteq \mathsf{Person}$ is known (in the T-box), one can deduce that: $\mathsf{Person}(\mathsf{Leonardo})$, $\mathsf{Person}(\mathsf{Vito})$ and then $\mathsf{Father}(\mathsf{Leonardo})$.

Given these primitive concepts and roles, it is possible to define many other related concepts:

$$\mathsf{Parent} \equiv \mathsf{Person} \sqcap \exists \mathsf{hasChild}.\mathsf{Person}$$

and

$$\mathsf{FatherWithoutSons} \equiv \mathsf{Male} \sqcap \exists \mathsf{hasChild}.\mathsf{Person} \ \sqcap \forall \mathsf{hasChild}.(\neg \mathsf{Male})$$

It is easy to see that the following relationships hold: $\mathsf{Parent} \sqsupseteq \mathsf{Father}$ and $\mathsf{Father} \sqsupseteq \mathsf{FatherWithoutSons}$. □

One of the most important inference services from the inductive learning viewpoint is *instance checking*, that is deciding whether an individual is an instance of a concept (w.r.t. an A-box). Related to this problem, it is often necessary to solve the *realization problem* that requires to compute, given an A-box and an individual the concepts which the individual belongs to:

Definition 3.2 (most specific concept).
Given an A-box $\mathcal{A}$ and an individual a, the most specific concept *of a w.r.t. $\mathcal{A}$ is the concept C, denoted $\mathsf{MSC}_{\mathcal{A}}(a)$, such that $\mathcal{A} \models C(a)$ and $\forall D$ such that $\mathcal{A} \models D(a)$, it holds: $C \sqsubseteq D$*
where $\models$ stands for the standard semantic deduction [CL73].

In the general case of a cyclic A-box expressed in a an expressive DL endowed with existential or numeric restriction the MSC cannot be expressed as a finite concept description [BCM$^+$03], thus it can only be approximated.

Since the existence of the MSC for an individual w.r.t. an A-box is not guaranteed or it is difficult to compute, generally an approximation of the MSC is considered up to a certain depth k. The maximum depth k has been shown to correspond to the depth of the considered A-box, as defined in [Man00].

Henceforth we will indicate generically an approximation to the maximum depth with MSC^*.

Especially for rich DL languages such as $\mathcal{ALC}$, many semantically equivalent (yet syntactically different) descriptions can be given for the same concept, which is the reason for preferring employing semantic approaches to reasoning over structural ones. Nevertheless equivalent concepts can be reduced to a normal form by means of rewriting rules that preserve their equivalence, such as: $\forall R.C_1 \sqcap \forall R.C_2 \equiv \forall R.(C_1 \sqcap C_2)$ (see [BCM+03] for issues related to normalization and simplification).

Particularly, an $\mathcal{ALC}$ normal form can be defined as follows:

Definition 3.3 ($\mathcal{ALC}$ normal form). *A concept description D is in $\mathcal{ALC}$* normal form *iff $D \equiv \bot$ or $D \equiv \top$ or if $D = D_1 \sqcup \cdots \sqcup D_n$ with*

$$D_i = \prod_{A \in \mathsf{prim}(D_i)} A \sqcap \prod_{R \in N_R} \left[\forall R.\mathsf{val}_R(D_i) \sqcap \prod_{E \in \mathsf{ex}_R(D_i)} \exists R.E \right]$$

where

- *for all $i = 1, \ldots, n$, $D_i \not\equiv \bot$*
- *$\mathsf{prim}(C)$ denotes the set of all (negated) concept names occurring at the top level of the description C;*
- *$\mathsf{val}_R(C)$ denotes the conjunction of concepts $C_1 \sqcap \cdots \sqcap C_n$ in the value restriction of role R, if any (otherwise $\mathsf{val}_R(C) = \top$);*
- *$\mathsf{ex}_R(C)$ denotes the set of concepts in the value restriction of the role R.*
- *for any R, every sub-description in $\mathsf{ex}_R(D_i)$ and $\mathsf{val}_R(D_i)$ is in normal form.*

This form can be employed for defining an ordering over the concept descriptions.

4 The Dissimilarity Measure

As a first step we need to define a dissimilarity measure for $\mathcal{ALC}$ descriptions. In order to achieve this goal, we introduce a function which is necessary for the correct definition of a dissimilarity measure. This should be a definite positive function on the set of $\mathcal{ALC}$ normal form concept description, defined making use of the syntax and semantics of the concepts (and roles) involved in the descriptions.

4.1 Overlap Function

The function is formally defined as follows:

Definition 4.1 (overlap function).
Let $\mathcal{L} = \mathcal{ALC}/_{\equiv}$ be the set of all concepts in $\mathcal{ALC}$ normal form and let $\mathcal{A}$ be an A-box with canonical interpretation $\mathcal{I}$ and let $|\Delta|$ be the number of all individuals

in the A-box. f *is a function*[1] $f : \mathcal{L} \times \mathcal{L} \mapsto R^+$ *defined as follows:*

for all $C, D \in \mathcal{L}$, *with* $C = \bigsqcup_{i=1}^{n} C_i$ *and* $D = \bigsqcup_{j=1}^{m} D_j$

$$f(C,D) := f_{\sqcup}(C,D) = \begin{cases} |\Delta| & \text{if } C \equiv D \\ 0 & \text{if } C \sqcap D = \bot \\ 1 + \lambda \cdot \max\limits_{\substack{i \in [1,n] \\ j \in [1,m]}} f_{\sqcap}(C_i, D_j) & \text{o.w.} \end{cases}$$

where λ *is a weighting factor*

$$f_{\sqcap}(C_i, D_j) := f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) + f_{\forall}(C_i, D_j) + f_{\exists}(C_i, D_j)$$

$$f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) := \frac{|PE(C_i) \cup PE(D_j)|}{|(PE(C_i) \cup PE(D_j)) \setminus (PE(C_i) \cap PE(D_j))|}$$

where,

- $PE(C_i) := (\bigsqcap_{P \in \mathsf{prim}(C_i)} P)^{\mathcal{I}}$ *and* $PE(D_j) := (\bigsqcap_{P \in \mathsf{prim}(D_j)} P)^{\mathcal{I}}$ *(extension of the primitive concepts conjunctions)*
- $f_P(\mathsf{prim}(C_i), \mathsf{prim}(D_j)) = |\Delta|$ *when* $(\mathsf{prim}(C_i))^{\mathcal{I}} = (\mathsf{prim}(D_j))^{\mathcal{I}}$

$$f_{\forall}(C_i, D_j) := \sum_{R \in N_R} f_{\sqcup}(\mathsf{val}_R(C_i), \mathsf{val}_R(D_j))$$

$$f_{\exists}(C_i, D_j) := \sum_{R \in N_R} \sum_{k=1}^{N} \max_{p=1,\ldots,M} f_{\sqcup}(C_i^k, D_j^p)$$

where $C_i^k \in \mathsf{ex}_R(C_i)$ *and* $D_j^p \in \mathsf{ex}_R(D_j)$ *and we suppose w.l.o.g. that* $N = |\mathsf{ex}_R(C_i)| \geq |\mathsf{ex}_R(D_j)| = M$, *otherwise the indices* N *and* M *are to be exchanged in the formula above.*

The function f represents a measure of the overlap between two concept descriptions (namely C and D) expressed in $\mathcal{ALC}$ normal form. It is defined recursively beginning from the top level of the concept descriptions (a disjunctive level) up to the bottom level represented by (conjunctions of) primitive concepts.

In case of disjunction, the overlap between the two concepts is equal to the maximum of the overlaps calculated among all couples of disjuncts (C_i, D_j) that make up the top level of the considered concepts, decreased by the weighting factor λ which might be defined as a function of the level where the (sub-concepts) occur within the overall concept descriptions (e.g. $\lambda = 1/\mathsf{level}$). This weight is useful to decrease the importance of the overlap of the sub-concepts; particularly the importance of the overlap decreases with the increasing of the level.

Then, since every disjunct is a conjunction of descriptions, it is necessary to calculate the overlap between conjunctive concepts. This overlap is calculated

[1] we omit the name $\mathcal{A}$ of the A-box for keeping the notation as simple as possible.

as the sum of the overlap measure calculated on the parts that make up the conjunctive concept description. Specifically, a conjunctive form can have three different types of terms: primitive concepts, universal restrictions and existential restrictions. Since concept conjunction ($\sqcap$) is a symmetric operator by definition, it is possible to put together every type of restriction (occurring at the same level) so it is possible to consider the conjunctions of primitive concepts ($\mathsf{prim}(C_i)$, $\mathsf{prim}(D_j)$), the conjunctions of existential restrictions and the conjunction of universal restrictions as specified in the definition of $\mathcal{ALC}$ normal form.

Next, the amount of the overlap for the three different type of conjunction is defined. Particularly, the amount of overlap between two conjunctions of (negated) primitive concepts is minimal if the they do not share any individual in their extension. Conversely, if the two concepts share some individual the overlap between them is computed as a measure of the intersection of their extensions with respect to their union.

The computation of the overlap between, respectively, concept descriptions expressed by universal and existential restrictions is a bit more complex. Considering the conjunction of universal restrictions, it is worthwhile to recall that every such restriction is a single conjunction linked by respect to a different role (since it is possible to write $\forall R.C \sqcap \forall R.D = \forall R.(C \sqcap D)$). Moreover, recall that the scope of each universal restriction is expressed in $\mathcal{ALC}$ normal form. Thus, the amount of the overlap between two concepts (within the disjuncts C_i and in D_j, resp.) that are scope of a universal restriction w.r.t. a certain role R is given by the amount of the overlap between two concepts expressed in $\mathcal{ALC}$ normal form (computed by $f_\sqcup$, as reported above); of course, if no disjunction occurs at the top level, it is possible to regard the concept description as a disjunction of single term to which $f_\sqcup$ applies in a simple way. Since we may have a conjunction of different concepts with universal restrictions, one per different role, the amount of the overlap of this conjunction is given by the sum of the overlap calculated for every universal restriction, rather than for every scope of a universal restriction. It is worth noting that, when a universal restriction on a certain role (say R) occurs in a disjunct (e.g. in C_i), but no such restriction on the same role occurs in the other description (say D_j), then we have that $\mathsf{val}_R(D_j) = \top$.

Now we turn to analyze the computation of the amount of the overlap between two descriptions made up of conjunctions of existential restrictions. For the dissimilarity between existential restrictions, we may recur the notion of *existential mapping* [KM01]. Let us suppose $N = |\mathsf{ex}_R(C_i)| \geq M = |\mathsf{ex}_R(D_j)|$. Such a mapping can be defined as a function:

$$\alpha : \{1, \ldots, N\} \mapsto \{1, \ldots, M\}$$

If each element of $\mathsf{ex}_R(C_i)$ and $\mathsf{ex}_R(D_j)$ is indexed with an integer in the ranges $[1, N]$ and $[1, M]$, respectively, then any function α maps each concept description $C_i^k \in \mathsf{ex}_R(C_i)$ with every descriptions $D_j^p \in \mathsf{ex}_R(D_j)$. Since each C_i^k (resp. D_j^p) is in $\mathcal{ALC}$ normal form, it is possible to calculate the amount of their overlap using $f_\sqcup$. Fixed a role R and considered a certain C_i^k (with $k \in [1, N]$), the amount of the overlap between C_i^k and D_j^p (with $p \in [1, M]$) is computed. We

are supposing that $N \geq M$, thus each existential restriction on role R is coupled with the one on the same role in other description scoring the maximum amount of overlap. These maxima are summed up per single role, then the sum is made also varying the role considered. In case of one role restriction on a certain role S is absent from either description then it is considered as the concept $\top$.

4.2 A Dissimilarity Measure

After clarifying the definition of f function, its meaning in all of its components, it is possible to derive a dissimilarity measure from f as shown in the following.

Definition 4.2 (dissimilarity measure). *Let $\mathcal{L}$ be the set of all concepts in normal form in $\mathcal{ALC}$ and let $\mathcal{A}$ be an A-box with canonical interpretation $\mathcal{I}$. The* dissimilarity measure *d is a function*

$$d : \mathcal{L} \times \mathcal{L} \mapsto [0, 1]$$

defined as follows:

for all $C = \bigsqcup_{i=1}^{n} C_i$ and $D = \bigsqcup_{j=1}^{m} D_j$ concept descriptions in $\mathcal{ALC}$ normal form, let

$$d(C, D) := \begin{cases} 1 & \text{if } f(C, D) = 0 \\ 0 & \text{if } f(C, D) = |\Delta| \\ 1/f(C, D) & \text{otherwise} \end{cases}$$

where f is the function defined above.

The function d measures the level of dissimilarity between two concept in $\mathcal{ALC}$ normal form using the function f that expresses the amount of overlap between the two concepts, say C and D. Particularly, if the overlap is minimal, i.e. $f(C, D) = 0$, then this means that there is no overlap between the considered concepts, therefore d must indicate that the two concepts are totally different, indeed $d(C, D) = 1$, i.e. it amounts to the maximum value of its range. If $f(C, D) = |\Delta|$ this means that the two concepts are totally overlapped and consequently $d(C, D) = 0$ that means that the two concept are undistinguishable, indeed d assumes the minimum value of its range. If the considered concepts have a partial overlap then their dissimilarity is lower as much as the two concept are more overlapped, since in this case $f(C, D) > 1$ and consequently $0 < d(C, D) < 1$.

An example is reported to clarify the usage of the dissimilarity measure:

Example 4.1. Let be C and D two concepts in $\mathcal{ALC}$ normal form and defined as follows:
$C \equiv A_2 \sqcap \exists R.B_1 \sqcap \forall T.(\forall Q.(A_4 \sqcap B_5)) \sqcup A_1$
$D \equiv A_1 \sqcap B_2 \sqcap \exists R.A_3 \sqcap \exists R.B_2 \sqcap \forall S.B_3 \sqcap \forall T.(B_6 \sqcap B_4) \sqcup B_2$
where A_i and B_j are all primitive concepts.

Now we calculate the amount of dissimilarity among the two concepts C and D; first, it is necessary to compute $f(C, D)$. Known that neither C nor D are

semantically equivalent nor are inconsistent, the value of f is estimated as in the third case of its definition. Let us denote with $C_1 := A_2 \sqcap \exists R.B_1 \sqcap \forall T.(\forall Q.(A_4 \sqcap B_5))$ and with $D_1 := A_1 \sqcap B_2 \sqcap \exists R.A_3 \sqcap \exists R.B_2 \sqcap \forall S.B_3 \sqcap \forall T.(B_6 \sqcap B_4)$. The computation of f is as follows

$$f(C,D) = \ 1 + \lambda \cdot \max\{\ f_{\sqcap}(C_1, D_1), f_{\sqcap}(C_1, B_2), f_{\sqcap}(A_1, D_1), f_{\sqcap}(A_1, B_2)\ \}$$

For brevity, we consider the computation of $f_{\sqcap}(C_1, D_1)$. $f_{\sqcap}$ is computed as the sum of f_P, $f_{\forall}$, $f_{\exists}$ i.e., respectively, f applied to primitive concepts, f applied to concepts in the universal restrictions, f applied to concepts in the existential restrictions.

Suppose that $(A_2)^{\mathcal{I}} \neq (A_1 \sqcap B_2)^{\mathcal{I}}$. Then:

$$\begin{aligned} f_P(C_1, D_1) &= f_P(\mathsf{prim}(C_1), \mathsf{prim}(D_1)) \\ &= f_P(A_2, \{A_1, B_2\}) \\ &= \frac{|PE(A_2) \cup PE(\{A_1, B_2\})|}{|(PE(A_2) \cup PE(\{A_1, B_2\})) \setminus (PE(A_2) \cap PE(\{A_1, B_2\}))|} \\ &= \frac{|(A_2)^{\mathcal{I}} \cup (A_1 \sqcap B_2)^{\mathcal{I}}|}{|((A_2)^{\mathcal{I}} \cup (A_1 \sqcap B_2)^{\mathcal{I}}) \setminus ((A_2)^{\mathcal{I}} \cap (A_1 \sqcap B_2)^{\mathcal{I}})|} \end{aligned}$$

In order to compute $f_{\forall}$ it is necessary to note that there are two roles at the same level (T and S), thus the summation over the different roles consists of two terms. Besides, the role S occurs only in D_1 and not in C_1, consequently $val_R(C_1) = \top$. Thus, in this case, we have:

$$\begin{aligned} f_{\forall}(C_1, D_1) = \sum_{R \in N_R} f_{\sqcup}(\mathsf{val}_{\mathsf{R}}(C_1), \mathsf{val}_{\mathsf{R}}(D_1)) = \\ = f_{\sqcup}(\mathsf{val}_{\mathsf{T}}(C_1), \mathsf{val}_{\mathsf{T}}(D_1)) + f_{\sqcup}(\mathsf{val}_{\mathsf{S}}(C_1), \mathsf{val}_{\mathsf{S}}(D_1)) = \\ = f_{\sqcup}(\forall Q.(A_4 \sqcap B_5), B_6 \sqcap B_4) + f_{\sqcup}(\top, B_3) \end{aligned}$$

The computation of $f_{\sqcup}(\forall Q.(A_4 \sqcap B_5), B_6 \sqcap B_4)$ and $f_{\sqcup}(\top, B_3)$ is the same reported above.

Now, by the definition of $f_{\exists}$, it is necessary to note that here is only a single role R, so the first summation collapses in one element. Then the number of conjunctive descriptions with existential restrictions w.r.t. the same role (S), are respectively $N = 2$ and $M = 1$, so we would have to find the max in a singleton, which is a simple case. So, we have:

$$f_{\exists}(C_1, D_1) = \textstyle\sum_{k=1}^{2} f_{\sqcup}(\mathsf{ex}_{\mathsf{R}}(C_1), \mathsf{ex}_{\mathsf{R}}(D_1^k)) = f_{\sqcup}(B_1, A_3) + f_{\sqcup}(B_1, B_2)$$

Also in this case, the computation of $f_{\sqcup}$ is similar to the case reported above.

In order to determine the dissimilarity value of C and D it is necessary to apply the same operations to the other elements and finally find the overlap. After that, the dissimilarity value can be computed straightforwardly by inverting the overlap.

4.3 Dissimilarity Measures for Individuals

The notion of *Most Specific Concept* MSC is commonly exploited for lifting individuals to the concept level. On performing experiments related to a similarity measure exclusively based on concept extensions [dFE05], we noticed that, resorting to the MSC, for adapting that measure to the individual to concept case, just falls short: indeed the MSCs may be too specific and unable to include other (similar) individuals in their extensions.

By comparing concept descriptions reduced to the normal form we have given a more structural definition of dissimilarity. However, since MSCs are computed from the same A-box assertions, reflecting the current knowledge state, this guarantees that structurally similar representations will be obtained for semantically similar concepts.

Let us recall that, given the A-box, it is possible to calculate the most specific concept of an individual a w.r.t. the A-box, $\mathsf{MSC}(a)$ (see Def. 3.2) or at least its approximation $\mathsf{MSC}^k(a)$ up to a certain description depth k. In the following we suppose to have fixed this k to the depth of the A-box, as shown in [Man00]. In some cases these are equivalent concepts but in general we have that $\mathsf{MSC}^k(a) \sqsupseteq \mathsf{MSC}(a)$.

Given two individuals a and b in the A-box, we consider $\mathsf{MSC}^k(a)$ and $\mathsf{MSC}^k(b)$ (supposed in normal form). Now, in order to assess the dissimilarity between the individuals, the d measure can be applied to these concept descriptions, as follows:

$$d(a, b) := d(\mathsf{MSC}^k(a), \mathsf{MSC}^k(b))$$

Analogously, the dissimilarity value between an individual a and a concept description C can be computed by determining the (approximation of the) MSC of the individual and then applying the dissimilarity measure:

$$\forall a: \; d(a, C) := d(\mathsf{MSC}^k(a), C)$$

These cases may turn out to be particularly handy in several tasks, namely both in inductive reasoning (construction, repairing of knowledge bases) and in information retrieval.

5 Discussion

In this section we intend to show that the function d presented in the previous section is really a dissimilarity measure and discuss some complexity issues related to its computation.

5.1 Properties of the Dissimilarity Measure

We prove that d function actually is a dissimilarity measure (or *dissimilarity function* [Boc99]), according to the following formal definition:

Definition 5.1 (dissimilarity measure). *Let* S *be a non empty set of elements. A* dissimilarity measure *for S is a real-valued function r defined on the set $S \times S$ that fulfills the following properties:*

1. $r(a, b) \geq 0 \quad \forall a, b \in S$ (positive definiteness)
2. $r(a, b) = r(b, a) \quad \forall a, b \in S$ (symmetry)
3. $\forall a, b \in S\colon\ r(a, b) \geq r(a, a)$

Proposition 5.1. *The function d is a dissimilarity measure for $\mathcal{L} = \mathcal{ALC}/_{\equiv}$.*

Proof.

1. (positive definiteness)
   trivial: by construction d computes dissimilarity by using sums of positive quantities and maxima computed on sets of such values.
2. (symmetry)
   by the commutativity of the sum and maximum operators.
3. $(\forall C, D \in \mathcal{L}\colon\ d(C, D) \geq d(C, C))$
   By the definition of d, it holds that $d(C, C) = 0$ and $d(C, C') = 0$ if C is semantically equivalent to C'. In all other different cases, $\forall D \in \mathcal{L}$ and D not semantically equivalent to D ($C \not\equiv D$), we have: $d(C, D) > 0$

5.2 Complexity Issues

The computational complexity of our dissimilarity measure d is strictly related to the computational complexity of the function f defined above. Besides, our measure relies on some reasoning services namely subsumption and instance-checking (see Sect. 3), therefore its complexity depends on the complexity of these inferences too. In order to define the complexity of d, we distinguish three different cases descending from being d based on the definition of f ($\mathit{Compl}(d) = \mathit{Compl}(f_{\sqcup})$) as in the following.

Let C, D be two concepts descriptions in normal form, with $C = \bigsqcup_{i=1}^{n} C_i$ and $D = \bigsqcup_{j=1}^{m} D_j$:

Case 1: *C and D are semantically equivalent.* In this case only subsumption is involved in order to verify the semantic equivalence of the concepts. Thus

$$\mathit{Compl}(d) = 2 \cdot \mathit{Compl}(\sqsupseteq)$$

where $\mathit{Compl}(\cdot)$ represents the complexity and $\sqsupseteq$ represents the subsumption operator

Case 2: *C and D are not semantically equivalent and C and D are disjoint.* In this case subsumption and conjunction are involved. Anyway, being the conjunction complexity constant in time, we have the same complexity of the previous case

Case 3: *C and D are not semantically equivalent and C and D are not disjoint.* In this case the complexity depends on the structure of the concepts involved. Particularly, it is necessary to calculate $f_\sqcap$ for $n \cdot m$ times; so the complexity is the following:

$$\begin{aligned} \mathit{Compl}(d) &= nm \cdot \mathit{Compl}(f_\sqcap) = \\ &= nm \cdot [\mathit{Compl}(f_P) + \mathit{Compl}(f_\forall) + \mathit{Compl}(f_\exists)] \end{aligned}$$

Thus it is necessary to analyze the complexity of f_P, $f_\forall$, $f_\exists$
In order to calculate f_P the most important operation is *Instance Checking* (IC) used for determining the concept extensions and there are two concepts involved. So we can conclude that:

$$C(f_P) = 2 \cdot C(IC)$$

The computation of $f_\forall$ and $f_\exists$ apply recursively the definition of $f_\sqcup$ on less complex descriptions. Particularly, $|N_R|$ calls of $f_\sqcup$ are needed for computing $f_\forall$ while the invocations of $f_\sqcup$ needed for $f_\exists$ are $|N_R| \cdot N \cdot M$, where $N = |\mathsf{ex}_R(C_i)|$ and $M = |\mathsf{ex}_R(D_j)|$ as in Def. 4.1. So we have that:

$$\mathit{Compl}(f_\forall) = |N_R| \cdot \mathit{Compl}(f_\sqcup)$$

$$\mathit{Compl}(f_\exists) = |N_R| \cdot M \cdot N \cdot \mathit{Compl}(f_\sqcup)$$

At this point we can sum up the complexity of this case as follows:

$$\begin{aligned} \mathit{Compl}(d) = nm \cdot [&(2 \cdot \mathit{Compl}(IC)) + \\ &+ (|N_R| \cdot \mathit{Compl}(f_\sqcup)) + \\ &+ (|N_R| \cdot M \cdot N \cdot \mathit{Compl}(f_\sqcup))] \end{aligned}$$

These considerations show that the complexity of the computation of d strongly depends on the complexity of the instance-checking for $\mathcal{ALC}$ which is P-space [BCM+03]. Nevertheless, in practical applications, these computations may be efficiently carried out exploiting the statistics that are maintained by the DBMSs query optimizers. Besides, the counts that are necessary for computing the concept extensions could be estimated by means of the probability distribution over the domain.

6 Conclusions and Further Developments

We have defined a measure of the overlap between concept descriptions expressed in $\mathcal{ALC}$ and then we have derived a dissimilarity measure and showed how to apply it to cases involving individuals.

Particularly the overlap function is based on both on the semantics and on the structure of the concepts involved. It is semantic because it is grounded on the concept extensions, as retrieved from current A-box. It is structural because

the measure is determined by computing the overlap of the sub-concepts nested in the considered concepts. The importance on the overlap depends on the level of the sub-concepts (in the normal form); this is expressed by the use of a weighting factor λ which should be a function of this level. Nevertheless the importance of primitive concepts and restrictions are different, therefore we are currently investigating on an extension where different typologies of sub-concepts are differently weighted.

Similarity and dissimilarity measures turn out to be useful in several applications and for many tasks such as commonality-based information retrieval in the context of terminological knowledge representation systems (which is a relatively new applicative context [Man00]), the realization of semantic search engines, classification, case-based reasoning, clustering, etc. In particular, This is our ultimate goal. A dissimilarity measure that is applicable both between concepts and between individuals and between concept and individual is suitable for both agglomerative and divisional clustering algorithms.

These ideas are being exploited also for defining other forms of similarity measures, namely kernels for relational representations like DLs, thus allowing the exploitation of the efficiency of the support vector machines, for example.

References

[BCM+03] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.

[BHP94] M. W. Bright, A. R. Hurson, and Simin H. Pakzad. Automated resolution of semantic heterogeneity in multidatabases. *ACM Transaction on Database Systems*, 19(2):212–253, 1994.

[Boc99] H.H. Bock. *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*. Springer-Verlag, 1999.

[BWH05] A. Borgida, T.J. Walsh, and H. Hirsh. Towards measuring similarity in description logics. In *Working Notes of the International Description Logics Workshop*, CEUR Workshop Proceedings, Edinburgh, UK, 2005.

[CH94] W.W. Cohen and H. Hirsh. Learning the CLASSIC description logic. In P. Torasso, J. Doyle, and E. Sandewall, editors, *Proceedings of the 4th International Conference on the Principles of Knowledge Representation and Reasoning*, pages 121–133. Morgan Kaufmann, 1994.

[CL73] C.L. Chang and R.C.T. Lee. *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, San Diego, 1973.

[dFE05] C. d'Amato, N. Fanizzi, and F. Esposito. A semantic similarity measure for expressive description logics. In A. Pettorossi, editor, *Proceedings of Convegno Italiano di Logica Computazionale (CILC05)*, Rome, Italy, 2005.

[GMV99] N. Guarino, C. Masolo, and G. Verete. Ontoseek: Content-based access to the web. *IEEE Intelligent Systems*, 3(14):70–80, 1999.

[JC97] J. Jang and D. Conrath. Semantic symilarity based on corpus statistic and lexical taxonomy. In *Proceedings of the International Conference on Computational Linguistics*, 1997.

[Kie02] J.-U. Kietz. Learnability of description logic programs. In S. Matwin and C. Sammut, editors, *Proceedings of the 12th International Conference on Inductive Logic Programming*, volume 2583 of *LNAI*, pages 117–132, Sydney, 2002. Springer.

[KM01] Ralf Kusters and Ralf Molitor. Computing least common subsumers in $\mathcal{ALEN}$. In B. Nebel, editor, *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI2001*, pages 219–224, 2001.

[Man00] T. Mantay. Commonality-based ABox retrieval. Technical Report FBI-HH-M-291/2000, Department of Computer Science, University of Hamburg, Germany, 2000.

[RE03] M.A. Rodríguez and M.J. Egenhofer. Determining semantic similarity among entity classes from different ontologies. *IEEE Transaction on Knowledge and Data Engineering*, 15(2):442–456, 2003.

[Res99] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[RV00] C. Rouveirol and V. Ventos. Towards learning in CARIN-$\mathcal{ALN}$. In J. Cussens and A. Frisch, editors, *Proceedings of the 10th International Conference on Inductive Logic Programming*, volume 1866 of *LNAI*, pages 191–208. Springer, 2000.

[SS04] S. Staab and R. Studer, editors. *Handbook on Ontologies*. International Handbooks on Information Systems. Springer, 2004.

[SSS91] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.

[Tve97] A. Tversky. Features od similarity. *Psycological Review*, 84(4):327–352, 1997.

[WB99] P. Weinstein and P. Bimingham. Comparing concepts in differentiated ontologies. In *Proceedings of 12th Workshop on Knowledge Acquisition, Modelling, and Management*, 1999.