A Similarity Measure for the $\mathcal{ALN}$ Description Logic

Nicola Fanizzi and Claudia d'Amato

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

Abstract. This work presents a similarity (and a derived dissimilarity) measure for Description Logics that are the theoretical counterpart of the standard representations for ontological knowledge. The focus is on the definition of a similarity measure for $\mathcal{ALN}$ concept descriptions, based both on the syntax and on the semantics of the descriptions elicited from the current state of the world. An extension of the measure is proposed for involving individuals and then for evaluating their (dis-)similarity, which makes it suitable for several (inductive) tasks.

1 Assessing the Similarity in Concept Languages

In the Semantic Web perspective [3], similarity plays an important role in several tasks, such as classification, clustering, retrieval and knowledge integration. Nevertheless, we are still at an initial phase in the definition of measures for assessing the similarity or the dissimilarity of concepts as described in the standard ontology languages [5].

Various distance measures for concept representations have been proposed in the literature (see a survey in [20]); they can be essentially categorized in three different types. *Path distance* measures have been defined as a function of the distance between terms in the hierarchical structure underlying an ontology [6]. The *feature matching* approach [24] uses both common and discriminant features among concepts and/or concept instances to compute the semantic similarity. Finally, there are methods founded on the *information content* [19, 10] where a similarity measure for concepts within a hierarchy is defined in terms of the variation of the information content conveyed by the concepts and the one conveyed by their immediate common super-concept. This is a measure of the variation of the information from a description level to a more general one.

Other measures compute the similarity among concepts belonging to different ontologies (e.g. see [25]). In [21] a similarity function determines similar classes by using a matching process making use of synonym sets, semantic neighborhood, and discriminating features that are classified into parts, functions, and attributes (see a recent survey in [23]). However, for the moment, this topic is beyond the scope of our work.

As pointed out in [5], most of the measures proposed so far are applicable to the assessment of the similarity of atomic concepts (within a hierarchy) rather

than on composite ones or they refer to very simple ontologies, built only using simple relations such as *is-a* and *part-of* (typical of lexical ontologies). Nevertheless, the standard ontology languages (e.g., OWL [18]) are founded in Description Logics (henceforth DLs) since they borrow the typical DLs constructors. Thus, it becomes necessary to investigate the similarity of more complex concept descriptions expressed in DLs. However, it has been observed that the structure of the descriptions becomes much less important when richer representations are adopted, due to the expressive operators that can be employed.

An approach intended for information retrieval purposes on DLs knowledge bases [16], aims at finding commonalities among concepts or among assertions, employing the *Least Common Subsumer* (LCS) operator [7] that computes the most specific generalization of the input concepts (with respect to subsumption). 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 measures defined in the cited works are suitable for very simple languages and not for the composite descriptions that can be obtained using the operators of DLs. Hence the semantics of these descriptions derives almost straightforwardly from their simple structures. We decided to focus our attention on measures which are essentially founded on semantics. Initially, we have defined dissimilarity measures between concept descriptions that virtually may work for any representation [9], being based exclusively on semantics. But this falls short when individuals come into play. Indeed, in the tasks which represent the final aim of our investigation on these measures, such as clustering, classification and retrieval, it is necessary to compute distances between individuals and concepts or between individuals. By recurring the notion of *most specific concept* (MSC) of an individual with respect to an ABox [1], measures based both on the concept structure and their semantics can be extended to such cases.

On the grounds of these ideas, we could define measures which are suitable for composite DLs descriptions and in particular for $\mathcal{ALC}$ [8, 10]. These measures elicit the underlying semantics by querying the knowledge base for assessing the information content of concept descriptions with respect to the knowledge base, as proposed also in [2]. In the perspective of defining a measure for more expressive ontology languages endowed with more constructors, with this work we intend to investigate and extend these ideas to languages endowed with numeric restrictions, starting from $\mathcal{ALN}$.

The remainder of this paper is organized as follows. In Sect. 2 the representation language $\mathcal{ALN}$ is presented. The similarity measure is illustrated and discussed in Sect. 3, with the extension to the cases involving individuals. Final remarks and possible applications and developments of the measure are examined in Sect. 4.

2 Background: The $\mathcal{ALN}$ Description Logic

$\mathcal{ALN}$ is a DLs language which allows for the expression of universal features and numeric constraints [1]. It has been adopted because of the tractability of the main related reasoning services [11]. Furthermore it has already been adopted also in other frameworks for learning in hybrid representations such as CARIN-$\mathcal{ALN}$ [22] or IDLP [13]. In order to keep this paper self-contained, syntax and semantics for the reference representation is briefly recalled with the characterization of the descriptions in terms of concept graphs.

2.1 Syntax and Semantics

In DLs, primitive *concepts* $N_C = \{A, \ldots\}$ are interpreted as subsets of a certain domain of objects and primitive *roles* $N_R = \{R, S, \ldots\}$ are interpreted as binary relations on such a domain. In $\mathcal{ALN}$, composite concept descriptions are built using atomic concepts and primitive roles by means of the constructors presented in Table 1. The meaning of such descriptions is defined by means of an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the *domain* of the interpretation and the functor $\cdot^{\mathcal{I}}$ (the *interpretation function*) maps concept and role descriptions to their extension: $\forall C \in N_C : C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and $\forall R \in N_R : R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

Table 1. Constructors and related interpretations for $\mathcal{ALN}$.

Name	Syntax	Semantics
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
primitive concept	A	$A^{\mathcal{I}} \subseteq \Delta$
primitive negation	$\neg A$	$\Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
concept conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
value restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y\ (x,y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
at-most restriction	$\leq n.R$	$\{x \in \Delta^{\mathcal{I}} \mid \ \lvert\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\}\rvert \leq n\}$
at-least restriction	$\geq n.R$	$\{x \in \Delta^{\mathcal{I}} \mid \ \lvert\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\}\rvert \geq n\}$

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. Differently from ILP, each (non primitive) concept has a single definition. Moreover, the DLs definitions are assumed not to be recursive, i.e. concepts cannot be defined in terms of themselves.

The A-box $\mathcal{A}$ 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 $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. Note that, differently from the examples in the ILP setting, the concept description C can be more complex than LP facts. For instance they could assert a universal property

of the an individual: $(\forall R.(A \sqcap \neg B))(a)$ that is, role R relates a exclusively to individuals[1] that are instances of the concept $A \sqcap \neg B$.

Example 2.1. Examples of $\mathcal{ALN}$ descriptions are [2]:

$$\begin{aligned}
\mathsf{Single} &\equiv \mathsf{Person} \sqcap \leq 0.\mathsf{marriedTo} \\
\mathsf{Polygamist} &\equiv \mathsf{Person} \sqcap \forall \mathsf{marriedTo}.\mathsf{Person} \sqcap \geq 2.\mathsf{marriedTo} \\
\mathsf{Bigamist} &\equiv \mathsf{Person} \sqcap \forall \mathsf{marriedTo}.\mathsf{Person} \sqcap = 2.\mathsf{marriedTo} \\
\mathsf{MalePolygamist} &\equiv \mathsf{Male} \sqcap \mathsf{Person} \sqcap \forall \mathsf{marriedTo}.\mathsf{Person} \sqcap \geq 2.\mathsf{marriedTo}
\end{aligned}$$

The notion of *subsumption* between DLs concept descriptions can be given in terms of the interpretations defined above:

Definition 2.1 (subsumption). *Given two concept descriptions C and D, C* subsumes *D iff it holds that $C^{\mathcal{I}} \supseteq D^{\mathcal{I}}$ for every interpretation $\mathcal{I}$. This is denoted denoted by $C \sqsupseteq D$. The induced equivalence relationship, denoted $C \equiv D$, amounts to $C \sqsupseteq D$ and $D \sqsupseteq C$.*

Note that this notion is merely semantic and independent of the particular DLs language adopted. It is easy to see that this definition also applies to the case of role descriptions.

A related inference used in the following is *instance checking*, that is deciding whether an individual is an instance of a concept [12, 1]. Conversely, it may be necessary to solve the *realization problem* that requires finding the concepts which an individual belongs to, especially the most specific one:

Definition 2.2. *Given an ABox $\mathcal{A}$ 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: $D \sqsupseteq C$.*

2.2 Structural Characterizations

Semantically equivalent (yet syntactically different) descriptions can be given for the same concept. However they can be reduced to a canonical form by means of equivalence-preserving rewriting rules, e.g. $\forall R.C_1 \sqcap \forall R.C_2 \equiv \forall R.(C_1 \sqcap C_2)$ (see [17, 1]). The normal form employs the notation needed to access the different parts (*sub-descriptions*) of a concept description C:

- $\mathsf{prim}(C)$ denotes the set of all (negated) concept names occurring at the top level of the description C;
- $\mathsf{val}_R(C)$ denotes conjunction of concepts $C_1 \sqcap \cdots \sqcap C_n$ in the value restriction of role R, if any (otherwise $\mathsf{val}_R(C) = \top$);
- $\mathsf{min}_R(C) = \max\{n \in \mathbb{N} \mid C \sqsubseteq (\geq n.R)\}$ (always a finite number);
- $\mathsf{max}_R(C) = \min\{n \in \mathbb{N} \mid C \sqsubseteq (\leq n.R)\}$ (if unlimited then $\mathsf{max}_R(C) = \infty$).

[1] It holds even in case no such $R-$filler is given.
[2] Here $(= n.R)$ is an abbreviation for $(\leq n.R \sqcap \geq n.R)$.

Definition 2.3 ($\mathcal{ALN}$ normal form). *A concept description C is in $\mathcal{ALN}$ normal form iff $C = \top$ or $C = \bot$ or*

$$C = \prod_{P \in \mathsf{prim}(C)} P \sqcap \prod_{R \in N_R} (\forall R.C_R \sqcap \geq n.R \sqcap \leq m.R)$$

where $C_R = \mathsf{val}_R(C)$, $n = \mathsf{min}_R(C)$ and $m = \mathsf{max}_R(C)$.

The complexity of normalization is polynomial [1]. Besides, subsumption can be checked in polynomial time too [11]. Note also that we are considering the case of subsumption with respect to empty terminologies that suffices for our purposes. Otherwise deciding this relationship may be computationally more expensive.

Although subsumption between concept descriptions is merely a semantic relationship, a more syntactic relationship can be found for a language of moderate complexity like $\mathcal{ALN}$ that allows for a structural characterization of subsumption [14].

Proposition 2.1 (subsumption in $\mathcal{ALN}$). *Given two $\mathcal{ALN}$ concept descriptions C and D in normal form, it holds that $C \sqsupseteq D$ iff all the following relations hold between the sub-descriptions:*

- $\mathsf{prim}(C) \subseteq \mathsf{prim}(D)$
- $\forall R \in N_R\colon\ \mathsf{val}_R(C) \sqsupseteq \mathsf{val}_R(D)$
- $\mathsf{min}_R(C) \leq \mathsf{min}_R(D) \wedge \mathsf{max}_R(C) \geq \mathsf{max}_R(D)$

Hence subsumption checking is accordingly polynomial like $O(n \log n)$, where n is the size of concept C. In the following we will refer to concepts descriptions in normal form unless a different case is explicitly stated.

The tree-structured representation of concept description are defined as follows [17]:

Definition 2.4 (description tree). *A* description tree *for a concept C in $\mathcal{ALN}$ normal form is a tree $\mathcal{G}(C) = (V, E, v_0, l)$ with root v_0 where:*

- *each node $v \in V$ is labelled with a finite set $l(v) \subseteq N_C \cup \{\neg A \mid A \in N_C\} \cup \{\geq n.R \mid n \in \mathbb{N},\ R \in N_R\} \cup \{\leq n.R \mid n \in \mathbb{N},\ R \in N_R\}$*
- *each edge in E is labelled with $\forall R$, where $R \in N_R$*

Proposition 2.2 (equivalence). *An $\mathcal{ALN}$ description C is semantically equivalent to an $\mathcal{ALN}$ description tree $\mathcal{G}(C)$ of size polynomial in the size of C, which can be constructed in polynomial time.*

Example 2.2. The concept description $D \equiv \forall R.(P \sqcap \forall S.Q) \sqcap \forall S.(Q \sqcap \leq 1S)$ is equivalent to the tree depicted in Fig. 1.

Instance checking can be characterized in terms of homomorphisms between trees and graphs standing for the ABoxes [17]:

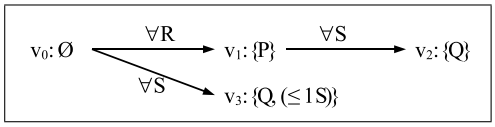

Fig. 1. The concept $D \equiv \forall R.(P \sqcap \forall S.Q) \sqcap \forall S.(Q \sqcap \leq 1S)$ as a description tree.

Definition 2.5 (A-box description graph). *Let $\mathcal{A}$ be an $\mathcal{ALN}$ A-box, a be an individual occurring in $\mathcal{A}$ ($a \in \mathsf{Ind}(\mathcal{A})$) and $C_a = \bigsqcap_{C(a)\in\mathcal{A}} C$. Let $\mathcal{G}(C_a) = (V_a, E_a, a, l)$ denote the description tree of C_a. $\mathcal{G}(\mathcal{A}) = (V, E, l)$ is a* A-box description graph *with:*

- $V = \bigcup_{a\in\mathsf{Ind}(\mathcal{A})} V_a$
- $E = \{aRb \mid R(a,b) \in \mathcal{A}\} \cup \bigcup_{a\in\mathsf{Ind}(\mathcal{A})} E_a$
- $l(v) = l_a(v) \quad$ *for all* $v \in V_a$

In a machine learning perspective, subsumption and instance checking can be used to translate an individual of the domain (an instance of the target concept) into a set of features suitable for propositional algorithms. Indeed, DLs that allow for efficient subsumption procedures, such as $\mathcal{ALN}$, are to be preferred.

3 Measure Definition

Using the structural notion of $\mathcal{ALN}$ normal form and the world-state as represented by the knowledge base, a similarity measure for the space of (equivalent) descriptions $\mathcal{L} = (\mathcal{ALN}\,|_{\equiv})$ can be defined as follows:

Definition 3.1 ($\mathcal{ALN}$ similarity measure). *The function $s : \mathcal{L} \times \mathcal{L} \mapsto [0,1]$ is inductively defined as follows. Given $C, D \in \mathcal{L}$:*

$$s(C,D) := \lambda \left[s_P(\mathsf{prim}(C), \mathsf{prim}(D)) + \frac{1}{|N_R|} \sum_{R\in N_R} s(\mathsf{val}_R(C), \mathsf{val}_R(D)) + \frac{1}{|N_R|} \sum_{R\in N_R} s_N((\mathsf{min}_R(C), \mathsf{max}_R(C)), (\mathsf{min}_R(D), \mathsf{max}_R(D))) \right]$$

where $\lambda \in]0,1]$ ($\lambda \leq 1/3$),

$$s_P(\mathsf{prim}(C), \mathsf{prim}(D)) := \frac{|\bigcap_{P_C\in\mathsf{prim}(C)} P_C^{\mathcal{I}} \cap \bigcap_{Q_D\in\mathsf{prim}(D)} Q_D^{\mathcal{I}}|}{|\bigcap_{P_C\in\mathsf{prim}(C)} P_C^{\mathcal{I}} \cup \bigcap_{Q_D\in\mathsf{prim}(D)} Q_D^{\mathcal{I}}|}$$

and if $\min(M_C, M_D) > \max(m_C, m_D)$ *then*

$$s_N((m_C, M_C), (m_D, M_D)) := \frac{\min(M_C, M_D) - \max(m_C, m_D) + 1}{\max(M_C, M_D) - \min(m_C, m_D) + 1}$$

else

$$s_N((m_C, M_C), (m_D, M_D)) := 0$$

The rationale for the measure is the following. Due to the relative simplicity of the language, the definition of operators working on $\mathcal{ALN}$ may be given structurally, as seen in the Sect. 2. Thus, we define the measure by recursively decomposing the normal form of the concept descriptions under comparison, and measure, per each level and separately, the similarity of the sub-concepts: primitive concepts, value restrictions, and number restrictions. We decided to combine the contribution of each similarity at a given level supposing a fixed[3] rate λ. Actually, in order to have the function s ranging over $[0, 1]$, λ should be less or equal to $1/3$.

The similarity of the primitive concept sets is computed as the ratio of the number of common individuals (belonging to both primitive conjuncts) with respect to the number of the individuals belonging to either conjunct. For those sub-concepts that are related through a role – say R – the similarity of the concepts made up by the fillers is computed recursively by applying the measure to $\mathsf{val}_R(\cdot)$. Finally, the similarity of the numeric restrictions is simply computed as a measure of the overlap between the two intervals. Namely it is the ratio of the amounts of individuals in the overlapping interval and those the larger one, whose extremes are minimum and maximum. Note that some intervals may be unlimited above: $\max = \infty$. In this case we may approximate with an upper limit N greater than $|\Delta| + 1$.

Note that the baseline of this measure is the extension of primitive concepts. Since such extensions cannot be known beforehand due to the *Open World Assumption* (OWA), we make an epistemic adjustment by assuming that it is approximated by retrieving[4] the concept instances based on the current world-state (i.e. according to the ABox $\mathcal{A}$):

$$P^I \leftarrow \{a \in \mathsf{Ind}(\mathcal{A}) \mid I \models_{\mathcal{A}} P(a)\}$$

The interpretation is not decisive because of the *unique names assumption* (UNA) holding for the individual names. Then, we may say that the *canonical interpretation*[5] [1] is considered for counting the retrieved individuals.

Furthermore, it can be foreseen that, per each level, before summing the three measures assessed on the three parts, these figures be normalized. Moreover, a lowering factor $\lambda_R \in]0, 1[$ may be multiplied so to decrease the impact of the sets of individuals related to the top-level ones through some role R.

Example 3.1 (computing the similarity). We show how the distance is practically computed on the ground of an ABox which can be supposed to have been completed according to the TBox descriptions (e.g. $\mathsf{Female} \sqsubseteq \neg\mathsf{Male}$).

[3] Actually we could assign different rates to the similarity of primitive concepts and numerical restrictions and the similarity of concepts for the role fillers.

[4] Formally, given the ABox $\mathcal{A}$ and a concept C, the retrieval service returns the individuals a such that $\mathcal{A} \models C(a)$.

[5] An interpretation where individual names occurring in the ABox stand for themselves.

Let such an ABox be

$$\mathcal{A} = \left\{ \begin{array}{l} \mathsf{Person(Meg)}, \neg\mathsf{Male(Meg)}, \mathsf{hasChild(Meg,Bob)}, \mathsf{hasChild(Meg,Pat)}, \\ \mathsf{Person(Bob)}, \mathsf{Male(Bob)}, \mathsf{hasChild(Bob,Ann)}, \\ \mathsf{Person(Pat)}, \mathsf{Male(Pat)}, \mathsf{hasChild(Pat,Gwen)}, \\ \mathsf{Person(Gwen)}, \neg\mathsf{Male(Gwen)}, \\ \mathsf{Person(Ann)}, \neg\mathsf{Male(Ann)}, \mathsf{hasChild(Ann,Sue)}, \mathsf{marriedTo(Ann,Tom)}, \\ \mathsf{Person(Sue)}, \neg\mathsf{Male(Sue)}, \\ \mathsf{Person(Tom)}, \mathsf{Male(Tom)} \end{array} \right\}$$

and let two descriptions be:

$$C \equiv \mathsf{Person} \sqcap \forall \mathsf{marriedTo}.\mathsf{Person} \sqcap \leq 1.\mathsf{hasChild}$$
$$D \equiv \mathsf{Male} \sqcap \forall \mathsf{marriedTo}.(\mathsf{Person} \sqcap \neg\mathsf{Male}) \sqcap \leq 2.\mathsf{hasChild}$$

Their similarity in the knowledge base is computed as follows (let $\lambda = 1/3$):

$$s(C,D) = \tfrac{1}{3} \cdot [s_P(\mathsf{prim}(C), \mathsf{prim}(D)) + \tfrac{1}{2} \textstyle\sum_{R \in N_R} s(\mathsf{val}_R(C), \mathsf{val}_R(D)) + \\ + \tfrac{1}{2} \textstyle\sum_{R \in N_R} s_N((\mathsf{min}_R(C), \mathsf{max}_R(C)), (\mathsf{min}_R(D), \mathsf{max}_R(D)))]$$

Now, we compute the three parts separately:

$$\begin{aligned} s_P(\mathsf{prim}(C), \mathsf{prim}(D)) &= s_P(\{\mathsf{Person}\}, \{\mathsf{Male}\}) = \\ &= \frac{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\} \cap \{\mathsf{Bob}, \mathsf{Pat}, \mathsf{Tom}\}|}{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\} \cup \{\mathsf{Bob}, \mathsf{Pat}, \mathsf{Tom}\}|} \\ &= \frac{|\{\mathsf{Bob}, \mathsf{Pat}, \mathsf{Tom}\}|}{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\}|} = 3/7 \end{aligned}$$

For the number restrictions on role $\mathsf{hasChild}$:

$$\begin{aligned} s_N((m_C, M_C), (m_D, M_D)) &= s_N((0,1),(0,2)) = \\ &= \frac{\min(1,2) - \max(0,0) + 1}{\max(1,2) - \min(0,0) + 1} = \frac{1 - 0 + 1}{2 - 0 + 1} = 2/3 \end{aligned}$$

For the number restrictions on role $\mathsf{marriedTo}$:

$$s_N((m'_C, M'_C), (m'_D, M'_D)) = 1$$

As regards the value restrictions on $\mathsf{marriedTo}$, $\mathsf{val}_{\mathsf{marriedTo}}(\mathsf{C}) = \mathsf{Person}$ and $\mathsf{val}_{\mathsf{marriedTo}}(\mathsf{D}) = \mathsf{Person} \sqcap \neg\mathsf{Male}$, hence:

$$s(\mathsf{Person}, \mathsf{Person} \sqcap \neg\mathsf{Male}) = 1/3 \cdot (s_P(\{\mathsf{Person}\}, \{\mathsf{Person}, \neg\mathsf{Male}\}) + 1 + 1)$$

and

$$\begin{aligned} s_P(\{\mathsf{Person}\}, \{\mathsf{Person}, \neg\mathsf{Male}\}) &= \frac{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\} \cap \{\mathsf{Meg}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}\}|}{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\} \cup \{\mathsf{Meg}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}\}|} \\ &= \frac{|\{\mathsf{Meg}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}\}|}{|\{\mathsf{Meg}, \mathsf{Bob}, \mathsf{Pat}, \mathsf{Gwen}, \mathsf{Ann}, \mathsf{Sue}, \mathsf{Tom}\}|} = 4/7 \end{aligned}$$

As there are no value restrictions on $\mathsf{hasChild}$, the similarity is maximal ($\mathsf{val}_{\mathsf{hasChild}}(\mathsf{C}) = \mathsf{val}_{\mathsf{hasChild}}(\mathsf{D}) = \top$).

Summing up:

$$\begin{aligned} s(C,D) &= \frac{1}{3}\left[\frac{3}{7} + \frac{1}{2}\left(\frac{1}{3}\left(\frac{4}{7}+1+1\right) + \frac{1}{3}\left(1+1+1\right)\right) + \frac{1}{2}\left(1+\frac{2}{3}\right)\right] \\ &= \frac{1}{3}\left[\frac{3}{7} + \frac{13}{14} + \frac{5}{6}\right] = \frac{92}{126} \simeq .7301 \end{aligned}$$

□

3.1 Discussion

It can be proven that s is really a similarity measure. (or *similarity function* [4]), according to the formal definition:

Definition 3.2 (similarity function). *Let S be a space of elements. A* similarity measure *f is a real-valued function defined on the set $S \times S$ that fulfills the following properties:*

1. $f(a,b) \geq 0 \quad \forall a,b \in S$ (positive definiteness)
2. $f(a,b) = f(b,a) \quad \forall a,b \in S$ (symmetry)
3. $\forall a,b \in S: \; f(a,b) \leq f(a,a)$

Proposition 3.1. *The function s is a similarity measure for the space $\mathcal{L}$.*

Proof. We have to prove the three properties:

1. *It is straightforward to see that s is positive definite since it is defined recursively as a sum of non-negative values.*
2. *s is also symmetric because of the commutativity of the operations involved, namely sum, minimum, and maximum (note that the value of s_N in Def. 3.1 does not change by exchanging C with D).*
3. *We must show that $\forall C,D \in \mathcal{L}: \; s(C,D) \leq s(C,C)$. This property can be proved by structural induction on D. The base cases are those related to primitive concepts and number restrictions, the inductive ones are those related to value restrictions and conjunctions:*
   - *if D is primitive then $s(C,D) = \lambda[s_P(\mathsf{prim}(C), \mathsf{prim}(D)) + s_1 + s_2] \leq$ $\lambda[\frac{|\bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}} \cap \bigcap_{Q_D \in \mathsf{prim}(D)} Q_D^{\mathcal{I}}|}{|\bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}} \cup \bigcap_{Q_D \in \mathsf{prim}(D)} Q_D^{\mathcal{I}}|} + 1 + 1] \leq$ $\lambda[\frac{|\bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}} \cap \bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}}|}{|\bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}} \cup \bigcap_{P_C \in \mathsf{prim}(C)} P_C^{\mathcal{I}}|} + 1 + 1] = \lambda[1+1+1] = s(C,C).$*
   - *if D is a number restriction the proof is analogous to the previous one, observing that* $0 \leq \frac{\min(M_C,M_D) - \max(m_C,m_D)}{\max(M_C,M_D) - \min(m_C,m_D)} \leq \frac{\min(M_C,M_C) - \max(m_C,m_C)}{\max(M_C,M_C) - \min(m_C,m_C)} \leq 1$

- *if D is a value restriction, then supposing by induction hypothesis that the property holds for descriptions whose depth is less than D's depth. This is the case of the sub-concept* $\mathsf{val}_R(D)$.
  Thus $s(\mathsf{val}_R(C), \mathsf{val}_R(D)) \leq s(\mathsf{val}_R(C), \mathsf{val}_R(C))$ *from which we may conclude that the property holds.*
- *if D is a conjunction of two simpler concepts, say* $\exists D_1, D_2 \in \mathcal{L} : D = D_1 \sqcap D_2$*, then assuming by induction hypothesis that the property holds for descriptions whose depth is less than D's depth such as* $D_{1,2}$*. This means that* $\forall i \in \{1, 2\} : s(C, D_i) \leq s(C, C)$*. It can be proven that* $\forall i \in \{1, 2\} : s(C, D) \leq s(C, D_i)$*. Hence the property holds.*

□

From a computational point of view, in order to control the computational cost of these functions, we may assume that the retrieval of the primitive concepts may be computed beforehand on the ground of the current knowledge base and then the similarity measure (or a derived dissimilarity measure) can be computed bottom-up through a procedure[6] based on dynamic programming.

3.2 Dissimilarity Measures Involving Individuals

Many machine learning algorithms (especially bottom-up ones) often require measuring the similarity between individuals. Also top-down ones are often based on a notion of *coverage* (instance checking) assessing the likelihood that an individual may belong to a concept by means of logic inferences or (somehow more simply) employing a notion of similarity between an individual and a concept description.

A dissimilarity measure can be easily derived from s in the following way:

Definition 3.3 ($\mathcal{ALN}$ dissimilarity measure). *The* dissimilarity function $d : \mathcal{L} \times \mathcal{L} \mapsto [0, 1]$ *is defined as follows. Given* $C, D \in \mathcal{L}$*:*

$$d(C, D) = 1 - s(C, D)$$

The notion of *Most Specific Concept* has been exploited for lifting individuals to the concept level [7]. On performing experiments related to a similarity measure exclusively based on concept extensions [9], 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 ABox assertions, reflecting the current knowledge state, this guarantees that structurally similar representations will be obtained for semantically similar concepts. In fact, in this way, all equivalent concepts written using the

[6] This procedure has been implemented for instance-based learning algorithms as well as the measures proposed in [8, 10].

same subconcepts but using different descriptions, can be expressed in the same form.

Let us recall that, given the ABox, it is possible to compute the most specific concept of an individual a w.r.t. the ABox, $\mathsf{MSC}(a)$ (see Def. 2.2) 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 ABox, as shown in [16]. 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 ABox, 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.

4 Final Remarks

Similarity and distance measures turn out to be useful in several tasks such as classification, case-based reasoning, clustering, etc. A novel (dis)similarity measure has been introduced, based on the information on concepts and roles as it can be approximated on the grounds of the underlying semantics of the ABox.

We have also shown how to apply this function to measuring the (dis)similarity between individuals and also between individual-concept (useful in knowledge discovery tasks). In particular, defining a measure, that is applicable for comparing both concepts and individuals, is suitable for agglomerative and divisional clustering. A further investigation will concern the derivation of a distance measure, which amounts to finding a measure that fulfils the triangular property.

The presented measure can be refined introducing a weighting factor, useful for decreasing the impact of the dissimilarity between nested sub-concepts in the descriptions on the determination of the overall value.

Another natural extension may concern the definition of dissimilarity measures in more expressive languages. For example, a normal form for $\mathcal{ALCN}$ can be obtained based on those for $\mathcal{ALN}$ and $\mathcal{ALC}$. Then, by exploiting the notion of existential mappings [15], already used for computing the LCS in $\mathcal{ALCN}$, the measure presented in this paper may be extended to the richer DL.

Kernels are another means to express the similarity in some unknown feature space. We are working at the definition of kernel functions on DLs representations, thus allowing the exploitation of the efficiency of kernel methods (e.g. support vector machines) in a relational setting.

References

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

[2] F. Bacchus. Lp, a logic for representing and reasoning with statistical knowledge. *Computational Intelligence*, 6:209–231, 1990.

[3] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.

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

[5] 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.

[6] 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.

[7] 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.

[8] C. d'Amato, N. Fanizzi, and F. Esposito. A dissimilarity measure for concept descriptions in expressive ontology languages. In H. Alani, C. Brewster, N. Noy, and D. Sleeman, editors, *Proceedings of the KCAP2005 Ontology Management Workshop*, Banff, Canada, 2005.

[9] 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. `http://www.disp.uniroma2.it/CILC2005/downloads/papers/15.dAmato_CILC05.pdf`.

[10] C. d'Amato, N. Fanizzi, and F. Esposito. A dissimilarity measure for $\mathcal{ALC}$ concept descriptions. In *Proceedings of the 21st Annual ACM Symposium of Applied Computing, SAC2006*, Dijon, France, 2006.

[11] F. M. Donini, M. Lenzerini, D. Nardi, and W. Nutt. The complexity of concept languages. *Information and Computation*, 134(1):1–58, 1997.

[12] F. M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Deduction in concept languages: From subsumption to instance checking. *Journal of Logic and Computation*, 4(4):423–452, 1994.

[13] 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.

[14] R. Küsters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. In F. Baader, G. Brewka, and T. Eiter, editors, *Proceedings of the Joint German/Austrian Conference on Artificial Intelligence, KI/ÖGAI01*, volume 2174 of *LNCS*, pages 33–47. Springer, 2001.

[15] R. Küsters and R. 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.

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

[17] R. Molitor. Structural subsumption for $\mathcal{ALN}$. Technical Report LTCS-98-03, LuFg Theoretical Computer Science, RWTH Aachen, Germany, 1998.

[18] OWL. Web Ontology Language Reference Version 1.0, 2003. `http://www.w3.org/TR/owl-ref`.

[19] 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.

[20] A. Rodriguez. *Assessing semantic similarity between spatial entity classes*. PhD thesis, University of Maine, 1997.

[21] 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.

[22] 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.

[23] P. Shvaiko and J. Euzenat. A survey of schema-based matching approaches. *Journal on Data Semantics*, IV:146–171, 2005.

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

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