Induction of Classifiers through Non-parametric Methods for Approximate Classification and Retrieval with Ontologies

NICOLA FANIZZI
Dipartimento di Informatica, Università degli studi di Bari
Via Orabona 4, 70125 Bari, Italy
fanizzi@di.uniba.it
http://www.di.uniba.it/~fanizzi

CLAUDIA d’AMATO
Dipartimento di Informatica, Università degli studi di Bari
Via Orabona 4, 70125 Bari, Italy
claudia.damato@di.uniba.it
http://www.di.uniba.it/~claudia.damato

FLORIANA ESPOSITO
Dipartimento di Informatica, Università degli studi di Bari
Via Orabona 4, 70125 Bari, Italy
esposito@di.uniba.it
http://www.di.uniba.it/~esposito

Received 24 September 2008
Revised Day Month Year
Accepted Day Month Year

This work concerns non-parametric approaches for statistical learning applied to the standard knowledge representations languages adopted in the Semantic Web context. We present methods based on epistemic inference that are able to elicit and exploit the semantic similarity of individuals in OWL knowledge bases. Specifically, a totally semantic and language independent semi-distance function is introduced, whence also an epistemic kernel function for Semantic Web representations is derived. Both the measure and the kernel function are embedded into non-parametric statistical learning algorithms customized for coping with Semantic Web representations. Particularly, the measure is embedded into a k-Nearest Neighbor algorithm and the kernel function is embedded in a Support Vector Machine. The realized algorithms are used to perform inductive concept retrieval and query answering. An experimentation on real ontologies proves that the methods can be effectively employed for performing the target tasks and moreover that it is possible to induce new assertions that are not logically derivable.

Keywords: Multi-relational Kernels; Nearest Neighbor Classification; Non-parametric Learning; Description Logics.
1. Learning from Ontologies

The Semantic Web (SW) represents an emerging domain where business, enterprise and organization on the Web will have its own organizational model (an ontology), and knowledge intensive automated manipulations on complex relational descriptions are foreseen [1]. Specific formal languages for knowledge representation have been designed for supporting a variety of applications in this context spanning from biological and geospatial fields to agents technology and service oriented architectures. Description Logics (DLs) [2], a family of languages that is endowed with well-founded semantics and reasoning services, have been adopted as the core technology of the ontology languages such as OWL.

Most of the research on formal ontologies has been focused on methods based on deductive reasoning. Yet, these methods may fail in case of noisy (and possibly inconsistent) data coming from heterogeneous sources. Then, inductive learning methods could be effectively employed to overcome this weakness. Nevertheless, the research on inductive methods and knowledge discovery applied to ontologic representations have received less attention [3, 4, 5, 6].

In this paper two non-parametric statistical learning approaches adapted for DL representations are proposed, the Nearest Neighbor (henceforth NN) approach [7] and the kernel methods [8] for the induction of classifiers that allow to perform important inferences on semantic knowledge bases (KBs), such as concept retrieval and query answering [2], in an alternative way. Indeed, such inferences can be cast as classification problems, related to assessing the class-membership of the individuals in the KB w.r.t. some query concepts.

Reasoning by analogy, similar individuals should likely belong to the extension of similar concepts. Moving from such an intuition, an instance-based framework (grounded on the mentioned inductive methods) has been devised for retrieving resources contained in ontological KBs, by inductively inferring (likely) consistent class-membership assertions that may not be logically derivable. As such, the resulting new (induced) assertions may enrich the KBs; indeed they can be suggested to the knowledge engineer that has only to validate them, thus making the ontology population task less burdensome [9].

Both the NN approach and kernel methods are known to be quite efficient and noise-tolerant and, differently from the other parametric statistical learning methods, they allow the precision of the hypothesis (the model to be learnt) to grow with the availability of data. Moreover, both are grounded on the exploitation of a notion of similarity. Specifically, the NN approach retrieves individuals belonging to query concepts, by analogy with the most similar training instances, namely on the grounds of the classification known for the nearest ones (w.r.t. a similarity or dissimilarity measure). Kernel methods represent a family of statistical learning algorithms, including the support vector machines (SVMs) [10], that can be very
efficient since they map, by means of a kernel function, the original feature space into a higher-dimensional space, where the learning task is simplified and where the kernel function implements a dissimilarity notion.

From a technical viewpoint, extending these inductive methods to a DL-based representation requires solutions for a number of issues: 1) a theoretical problem is posed by the Open World Assumption (OWA) that is generally made on the semantics of the DL representation adopted for the ontologies, differently from the typical database settings, where the Closed World Assumption (CWA) is made; 2) both NN and kernel methods are devised for classification problems where classes are assumed to be pairwise disjoint. This is quite unlikely in the SW context where this information is often overlooked and an individual can be instance of more than one concept in the hierarchy represented by the ontology; 3) suitable metrics, namely (dis-)similarity measures and kernel functions, are necessary for coping with the high expressive power of DL representations; such definitions could not be straightforward.

As discussed in [11], most of the existing measures focus on concept similarity and particularly on the similarity of atomic concepts within hierarchies or simple ontologies. Conversely, for our purposes, a notion of dissimilarity between single objects (individuals in the DL terminology) is required. Recently, dissimilarity measures for specific DL concept descriptions have been proposed [12, 13]. Although they turned out to be quite effective for the inductive tasks of interest, they are partly based on structural criteria (a notion of normal form) which determine their main weakness: they are hardly scalable to deal with the complex DL languages backing the standard ontology languages.

To overcome these limitations, new totally semantic pseudo-metrics [14] can be exploited. These language-independent measures assess the dissimilarity of two individuals on the ground of their behavior w.r.t. a given context [15] which can be represented by a committee of features (concepts), selected from those defined in the KB or that can be generated to this purpose. Since also kernel functions implement a notion of similarity, we have derived a family of kernel functions from the semantic dissimilarity measures. As for the function from which it is derived, the kernel is language independent and it is based on the semantics of the individual as epistemically elicited from the KB w.r.t. a number of dimensions, represented by a committee of discriminant concept descriptions (features).

The pseudo-metrics and the kernel functions have been integrated, respectively, within a NN classification algorithm and with a kernel machine (specifically a SVM). The resulting system has been used for performing inductive concept retrieval and query answering with both approaches. Experimentally we show that the methods

\[b\] The choice of optimal committees may be performed in advance through randomized search algorithms [14, 16].

\[c\] As for the pseudo-metrics [14], also for the kernel function, the feature set can be optimally generated by means of a simulated annealing procedure [14] or other stochastic optimization methods [16].
based on the induced classifiers can perform concept retrieval and query answering comparably well w.r.t. a standard deductive reasoner. Even more so, the proposed procedure is also sometimes able to induce new concept assertions that are not logically derivable, namely when the reasoner can return an unknown classification reply on class-membership queries because it has no sufficient evidence for assigning an individual to a certain concept or to its negation.

The paper is organized as follows. In the next section the basics of the reference representation are briefly summarized. In Sect. 3 the basics of the NN approach and its extension to the SW setting is analyzed and the family of pseudo-metrics used for performing the NN search is presented. In Sect. 4 the basics of kernel methods and kernel functions are recalled with the definition of a family of kernel functions derived from the pseudo-metrics. In Sect. 5 the experimental evaluation of the proposed methods on a number of OWL ontologies is discussed. Conclusions are drawn in Sect. 6.

2. Representation and Inference

The basics of the DLs will be briefly recalled. The reader may refer to the DL handbook [2] for a thorough reference. Such representations provide the standard language constructors adopted by the standard ontology languages employed in the SW, such as OWL-DL. We first illustrate in detail \( \mathcal{ALC} \) logic and then extend the representation to more expressive decidable fragments.

2.1. Knowledge Bases in Description Logics

As for the building blocks of the language, let us consider a triple \((N_C, N_R, N_I)\) made up, respectively, of a set of primitive concept names \(N_C\), a set of primitive role names \(N_R\) and a set of individual names \(N_I\). The semantics of these symbols is defined by an interpretation \(\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})\), where \(\Delta^\mathcal{I}\) is a non-empty set, the domain of the interpretation, and \(\cdot^\mathcal{I}\) is the interpretation function that maps each individual \(a \in N_I\) to a domain object \(a^\mathcal{I}\), each primitive concept name \(A \in N_C\) to its extension \(A^\mathcal{I} \subseteq \Delta^\mathcal{I}\) and, dually, for each \(R \in N_R\) the extension is a binary relation \(R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}\). Special symbols denote, respectively, the top concept \(\top\), interpreted as the whole domain \(\Delta^\mathcal{I}\), and the bottom concept \(\bot\) corresponding to an empty extension \(\emptyset\).

The interpretations naturally extend to concept descriptions that can be built using a number of constructors\(^c\). The language supports full negation: given any concept description \(C\), denoted \(\neg C\), it amounts to \(\Delta^\mathcal{I} \setminus C^\mathcal{I}\). The conjunction of concepts, denoted with \(C_1 \sqcap C_2\), whose extension is \(C_1^\mathcal{I} \cap C_2^\mathcal{I}\) and, dually, concept disjunction, denoted with \(C_1 \sqcup C_2\), whose extension is \(C_1^\mathcal{I} \cup C_2^\mathcal{I}\). Finally, there are some

\(^c\)Although n-ary extensions have been proposed as well [2].

\(^d\)In fact, the \(\mathcal{ALC}\) DL corresponds to the fragment of first-order logic obtained by restricting the syntax to formulae containing two variables. \(\mathcal{ALC}\) has a modal logic counterpart, namely the multi-modal version of the logic \(K\) [2].
restrictions on roles: the existential restriction, denoted with $\exists R.C$, and interpreted as the set \( \{ x \in \Delta^I \mid \exists y \in \Delta^I : (x, y) \in R^I \land y \in C^I \} \) and the value restriction, denoted with $\forall R.C$, whose extension is \( \{ x \in \Delta^I \mid \forall y \in \Delta^I : (x, y) \in R^I \rightarrow y \in C^I \} \).

The main inference employed with these representations is assessing whether a concept subsumes another concept based on their semantics:

**Definition 1.** (subsumption) Given two descriptions $C$ and $D$, $C$ is subsumed by $D$, denoted by $C \sqsubseteq D$, iff for every interpretation $I$ it holds that $C^I \subseteq D^I$.

When $C \sqsubseteq D$ and $D \sqsubseteq C$ then they are equivalent, denoted with $C \equiv D$.

This relation can be defined also on roles (whose extension can be regarded as a set of couples) and is often restricted to the interpretations satisfying the knowledge bases (see below).

Further constructors have been proposed to extend the expressiveness of this language. We will be interested to the language that constitute the counterpart of OWL, namely $SHO\Omega Q(D)$ that, roughly, extends $ALC$ with transitive roles, role hierarchies (admitting subsumption axioms between roles like $\text{hasDaughter} \sqsubseteq \text{hasChild}$), individual classes (concepts defined by enumerating their individuals, e.g. \{TOM, ANN\}), inverse roles ($\text{hasChild}^{-}$ should correspond to $\text{hasParent}$), and qualified number restrictions (e.g. parents with at most 3 children: $\leq 3.\text{hasChild.Human}$). Besides concrete domains$^f$ (D) can be imported with their own semantics to represent other data-types.

**Definition 2.** (knowledge base) A knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. $\mathcal{T}$ is the set of terminological axioms of concept descriptions $C \equiv D$, where $C$ is the concept name and $D$ is its description. $\mathcal{A}$ contains assertions on the world state, e.g. $C(a)$ and $R(a, b)$, meaning that $a^I \in C^I$ and $(a^I, b^I) \in R^I$.

Note that defined concepts must have a unique definition. However more general axioms defining concepts through inclusions ($C \sqsubseteq D$) may be also admitted. Moreover, such definitions are assumed not to be recursive$^g$.

Note that the Open World Assumption is made on the underlying semantics, since normally knowledge representation systems are applied in situations where one cannot assume that the knowledge in the KB is complete. This is convenient for the Semantic Web context where new resources (e.g. Web pages, Web services) may be continuously made available.

**Example 1.** (Open-world semantics) A simple knowledge base modeling concepts and roles related to the British royal family$^h$ may be crafted as follows:

\[
\mathcal{T} = \{ \text{Male} \equiv \neg \text{Female} , \}
\]

---

$^f$Concrete domains include data types with their own semantics, such as numerical types, but also more elaborate domains, such as tuples of the relational calculus, spatial regions, or time intervals.

$^g$Recursive concept definitions require more complex fixpoint semantics [2].

$^h$From Franconi’s DLs course: [http://www.inf.unibz.it/~franconi/dl/course](http://www.inf.unibz.it/~franconi/dl/course)
Woman ≡ Human ⊓ Female,
Man ≡ Human ⊓ Male,
Mother ≡ Woman ⊓ ∃hasChild.¬Human,
Father ≡ Man ⊓ ∃hasChild.¬Human,
Parent ≡ Father ⊔ Mother,
Grandmother ≡ Woman ⊓ ∃hasChild.¬Parent,
Mother-w/o-daughter ≡ Mother ⊓ ∀hasChild.¬Female,
Super-mother ≡ Mother ⊓ ≥ 3. hasChild 

A = { Woman(elisabeth), Woman(diana),
       Man(charles), Man(edward), Man(andrew),
       Mother-w/o-daughter(diana),
       hasChild(elisabeth, charles), hasChild(elisabeth, edward),
       hasChild(elisabeth, andrew), hasChild(diana, william), hasChild(charles, william) }

Given such a KB, one may infer assertions like Father(charles), Mother(diana) or Super-mother(elisabeth) but not Mother-w/o-daughter(elisabeth) because it may well be that a daughter is not known yet.

2.2. Inference Services

The basic inference services for KBs amount to verifying whether a certain relationship is a logical consequence of the axioms contained in the KB. Typical reasoning tasks for KBs require to determine whether a concept description is satisfiable (i.e., it admits a model), or whether it subsumes another one (w.r.t. the KB models). Reasoning is generally implemented through tableau algorithms [2].

Besides checking whether the set of ABox assertions is consistent w.r.t. the KB models, ABox reasoning may require making further non-standard inferences [2]. Since we aim at crafting inductive methods that manipulate single individuals, an important related inference is instance checking, that amounts to deciding whether an individual is an instance of a given concept. This view has consequences for the way inference service requests are answered. Note that, because of the OWA, a reasoner might be unable to answer certain requests.

Thus it may happen that an object that cannot be proved to belong to a certain concept is not necessarily a counterexample for that concept. That would only be interpreted as a case of insufficient (incomplete) knowledge for that assertion.

Another related inference is retrieval which consists in finding the individuals that belong to a given concept:

Definition 3. (retrieval) Given an knowledge base K and a concept C, find all individuals \( a \in \text{Ind}(A) \) such that \( K \models C(a) \).

[1] Models could be constructed for both the membership and non-membership case [2].
[2] In the following \( \text{Ind}(A) \) denotes the set of individuals occurring in the ABox \( A \).
A straightforward algorithm for a retrieval query can be realized by checking whether each individual occurring in the ABox may be an instance of the given concept $C$.

3. Query Answering as Nearest Neighbor Search

Query answering boils down to determining whether a resource belongs to a (query) concept extension. Here, an alternative inductive method is proposed for retrieving the resources that likely belong to a query concept. Such a method may also be able to provide an answer even when it may not be inferred by deduction.

In similarity search [17] the basic idea is to find the most similar object(s) to a query one (i.e. the one that has to be classified) w.r.t. a similarity (or dissimilarity) measure. We review the basics of the NN method applied to the SW context [13].

3.1. The method

From a Machine Learning viewpoint, this method can be ascribed to the category of lazy learning, since the learning phase is reduced to memorizing instances of the target concepts pre-classified by an expert. Then, during the classification phase, a notion of similarity over the instance space is employed to classify a new instance in analogy with its neighbors.

The objective is to induce an approximation for a discrete-valued target hypothesis function $h : IS \mapsto V$ from a space of instances $IS$ to a set of values $V = \{v_1, \ldots, v_s\}$ standing for the classes (concepts) that have to be predicted. In order to do this it is assumed that the correct value for $h$ is known for a limited set of training instances $TIS$ such that $|TIS| \ll |IS|$, i.e. only a limited number of training instances is needed.

Let $x_q$ be the query instance whose class-membership is to be determined. Using a dissimilarity measure, the set of the $k$ nearest (pre-classified) training instances w.r.t. $x_q$ is selected: $NN(x_q) = \{x_i \mid i = 1, \ldots, k\}$. A $k$NN algorithm approximates $h(x_q)$ on the grounds of the value that $h$ assumes for the training instances in $NN(x_q)$, i.e. the $k$ closest instances to $x_q$. More precisely, the value is decided by means of a weighted majority voting procedure, namely the class-value in $V$ is the most voted one by the training instances in $NN(x_q)$, where each vote is also weighted by the similarity of the neighbor instance. Formally, the estimation of the hypothesis function for the query individual is defined as:

$$\hat{h}(x_q) := \arg \max_{v \in V} \sum_{i=1}^{k} w_i \delta(v, h(x_i))$$

where $\delta$ returns 1 in case of matching arguments and 0 otherwise, and, given a dissimilarity measure $d$, the weights are determined by $w_i = 1/d(x_i, x_q)$.

Note that the estimate function $\hat{h}$ is defined extensionally: the basic NN does not return an intensional classification model (i.e. a function or a concept definition), it
is merely able to give an answer for the instances to be classified. It should be also observed that this setting assigns a value to the query instance which stands for one class in a set of pairwise disjoint classes (corresponding to the value set $V$). In a multi-relational setting such as the SW context this assumption cannot be made in general since an individual may be instance of more than one concept. Hence the overall learning problem should be decomposed in a number of simpler binary ones, focusing on assessing the membership (or non-membership) of instances to a given query concept $Q$.

Another issue is represented by the CWA usually made in machine learning and knowledge discovery, as opposite to the OWA characterizing the SW context. To deal with the OWA, the absence of information on whether a training instance $x$ belongs to the extension of the query concept $Q$ should not be interpreted negatively, as in the settings which adopt the CWA. Rather, it should count as neutral information. Thus, assuming the alternate viewpoint, the multi-class classification problem is transformed into a ternary one. Hence another value set has to be adopted, namely $V = \{+1, -1, 0\}$, where the three values denote membership, non-membership, and the unknown classification case, respectively.

The task can be cast as follows: given a query concept $Q$, determine the membership of an instance $x_q$ through the NN procedure (see Eq. 1) where $V = \{-1, 0, +1\}$ and the hypothesis function values for the training instances are determined as follows:

$$h_Q(x) = \begin{cases} +1 & \mathcal{K} \models Q(x) \\ -1 & \mathcal{K} \models \neg Q(x) \\ 0 & \text{otherwise} \end{cases}$$

i.e. the value of $h_Q$ for the training instances is determined by the entailment of the corresponding assertion from the knowledge base.

Note that, being this procedure based on a majority vote of the individuals in the neighborhood, it is less error-prone in case of noise in the data (e.g. incorrect assertions) w.r.t. a purely logic deductive procedure. Therefore it may be able to give a correct classification even in case of inconsistent knowledge bases.

### 3.2. Measuring the Likelihood of an Answer

At the same time, it should be noted that the inductive inference made by the procedure shown above is not guaranteed to be deductively valid. Indeed, inductive inference naturally yields a certain degree of uncertainty.

In order to measure the likelihood of the decision made by the procedure, given the nearest training individuals in $NN(x_q, k) = \{x_1, \ldots, x_k\}$, the quantity that determined the decision can be normalized by dividing it by the sum of such arguments over the (three) possible values:

\(^{\text{We use } \models \text{ to denote entailment, as computed through a reasoner.}}\)
\[ l(\text{class}(x_q) = v|NN(x_q,k)) = \frac{\sum_{i=1}^{k} w_i \cdot \delta(v,h_Q(x_i))}{\sum_{v' \in V} \sum_{i=1}^{k} w_i \cdot \delta(v',h_Q(x_i))} \]  

(2)

Hence the likelihood of the assertion \( Q(x_q) \) corresponds to the case when \( v = +1 \).

The evaluation of the likelihood of the reply of the inductive classifier may have several applications, such as it can be employed for ranking the query hits (i.e. those resources whose assertions approximately satisfy the query).

3.3. A Semantic Pseudo-Metric for Individuals

Various attempts to define semantic similarity (or dissimilarity) measures for concept languages have been made, yet they had still a limited applicability to simple languages [11] or they are not completely semantic depending also on the structure of the descriptions [13].

Moreover, for our purposes, a function that can measure the similarity of individuals is needed rather than one for concepts. It can be observed that individuals do not have a syntactic structure that can be compared. This has led to lifting them to the concept description level before comparing them resorting to the notion of the most specific concept of an individual w.r.t. the ABox [2], which makes the measure language-dependent. Besides, it would add a further approximations as the most specific concepts can be defined only for simple DLs. Conversely, we intend to exploit a dissimilarity measure that totally depends on semantic aspects of the individuals in the knowledge base.

Following the ideas borrowed from [18], totally semantic distance measures for individuals can be defined w.r.t. a KB. The resulting measure is based on the idea of comparing the semantics of the input individuals along a number of dimensions represented by a committee of concept descriptions which may be regarded as the context of the comparison [15]. Indeed, on a semantic level, similar individuals should behave similarly w.r.t. the same concepts.

More formally, the rationale is to compare individuals on the grounds of their semantics w.r.t. a collection of concept descriptions which stands as a group of discriminating features expressed in the fragment of OWL-DL language taken into account. In its simple formulation, a family of distance functions for individuals inspired to Minkowski’s norms \( L_p \) can be defined as follows [14]:

Definition 4. (family of dissimilarity measures) Let \( \mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle \) be a KB. Given a set \( \mathcal{F} = \{F_1, F_2, \ldots, F_m\} \) of concept descriptions, a family of dissimilarity functions

\[ d_p^\mathcal{F} : \text{Ind}(\mathcal{A}) \times \text{Ind}(\mathcal{A}) \mapsto [0,1] \]

is defined as follows:
∀(a, b) ∈ Ind(\mathcal{A}) \times Ind(\mathcal{A})

\[ d^\mathcal{F}_p(a, b) := \frac{1}{|\mathcal{F}|} \left( \sum_{i=1}^{|\mathcal{F}|} w_i | \delta_i(a, b) |^p \right)^{1/p} \]

where \( p > 0 \) and \( \forall i \in \{1, \ldots, m\} \) the dissimilarity function \( \delta_i \) is defined:

\[ \forall(a, b) \in Ind(\mathcal{A}) \times Ind(\mathcal{A}) \]

\[ \delta_i(a, b) = \begin{cases} 
0 & (F_i(a) \in \mathcal{A} \land F_i(b) \in \mathcal{A}) \lor (\neg F_i(a) \in \mathcal{A} \land \neg F_i(b) \in \mathcal{A}) \\
1 & (F_i(a) \in \mathcal{A} \land \neg F_i(b) \in \mathcal{A}) \lor (\neg F_i(a) \in \mathcal{A} \land F_i(b) \in \mathcal{A}) \\
1/2 & \text{otherwise} 
\end{cases} \]

or, model theoretically:

\[ \forall(a, b) \in Ind(\mathcal{A}) \times Ind(\mathcal{A}) \]

\[ \delta_i(a, b) = \begin{cases} 
0 & (K \models F_i(a) \land K \models F_i(b)) \lor (K \models \neg F_i(a) \land K \models \neg F_i(b)) \\
1 & (K \models F_i(a) \land K \models \neg F_i(b)) \lor (K \models \neg F_i(a) \land K \models F_i(b)) \\
1/2 & \text{otherwise} 
\end{cases} \]

The model theoretic definition for the projections, requires the entailment of an assertion (instance-checking) rather than the simple ABox look-up; this can make the measure more accurate yet more complex to compute unless a KBMS is employed that maintains such information at least for the concepts in \( \mathcal{F} \).

It can be proved [14] that these functions have the standard properties for pseudo metrics (i.e. semi-distances [17]). This means that it cannot be proved that \( d^\mathcal{F}_p(a, b) = 0 \) iff \( a = b \) (indiscernible case). This is the case of indiscernible individuals with respect to the given set of features \( \mathcal{F} \). However several methods have been proposed for avoiding this case [14] involving the consideration of equivalent classes of individuals or the adoption of a supplementary meta-feature \( F_0 \) determining the equality of the two individuals: \( \delta_0(a, b) = 0 \) if \( a^F = b^F \) otherwise \( \delta_0(a, b) = 1 \).

Compared to other proposed dissimilarity measures [11, 13], the presented functions do not depend on the constructors of a specific language, rather they require only (retrieval or) instance-checking for computing the dissimilarity functions \( \delta_i \) through class-membership queries to the knowledge base.

The computational complexity of measuring the dissimilarity of two individuals depends on the complexity of such inferences (see [2], Ch. 3). Note also that the \( \delta_i \) functions that determine the measure can be computed (or derived from statistics maintained on the knowledge base) before the actual distance application, thus determining a speed-up in the computation of the measure. This is very important for algorithms that massively use this distance, such as all instance-based methods.

The measures strongly depend on \( \mathcal{F} \). Here, the assumption that the \( \mathcal{F} \) represents a sufficient number of (possibly redundant) features that are able to discriminate really different individuals is implicitly made. Anyway, optimal feature can be learnt by the use of a randomized optimization procedure [14]. Nevertheless, it has been experimentally shown that good results could be obtained by using the very set of both primitive and defined concepts in the KB.
Of course these approximate measures become more and more precise as the knowledge base is populated with an increasing number of individuals.

4. Kernel Methods

Kernel methods [8] represent a family of statistical learning algorithms (including the support vector machines [10]) that have been effectively applied to a variety of tasks, recently also in domains that typically require structured representations [19, 20]. They can be very effective since they map, by means of a kernel function, the original feature space into a high-dimensional space, where learning a linear classifier with a suitable method is possible. Such a mapping is not explicitly performed (kernel trick): the usage of a positive definite kernel function (i.e. a valid kernel) ensures that the embedding into a new space exists and that the kernel function corresponds to the inner product in this space [8].

4.1. Kernels for Structured Representations

From an engineering viewpoint, two components of the kernel methods have to be distinguished: the kernel machine and the kernel function: the former encapsulates the learning task, while the latter encodes the hypothesis language. In this way, an efficient algorithm for attribute-value instance spaces can be applied to structured spaces (e.g. trees, graphs) by merely replacing the kernel function. This motivates the attractiveness of the SVMs and other kernel methods [8] as they can reproduce learning in high-dimensional spaces while working efficiently in a vectorial representation.

Kernels functions are endowed with the closure property w.r.t. many operations, one of them is the convolution [21]: kernels can deal with compounds by decomposing them into their parts, provided that valid kernels have already been defined for them.

\[ k_{\text{conv}}(x, y) = \sum_{\bar{x} \in R^{-1}(x)} \prod_{i=1}^{D} k_i(x_i, y_i) \]  

where \( R \) is a composition relationship building a single compound out of \( D \) simpler objects, each from a space that is already endowed with a valid kernel. The choice of the function \( R \) is a non-trivial task which may depend on the particular application.

On the ground of this property several kernel functions have been defined: for strings, trees, graphs and other discrete structures [19]. In [20], generic kernels based on type construction are formalized, where types are declaratively defined. In [22], kernels parametrized on a uniform DL representation are introduced. Specifically, a syntax-driven kernel definition, based on a simple DL representation (the Feature Description Language), is given. Kernel functions for structured data, parametrized on a description language, allow for the employment of algorithms such as SVMs
that can simulate feature generation. These functions transform the initial representation of the instances into the related active features, thus allowing learning the classifier directly from the structured data.

Kernel functions for the SW representations have also been defined. Specifically, in [23] a kernel for comparing ALC concept definitions is introduced. It is based on the structural similarity of the AND-OR trees corresponding to the normal form of the input concepts. This kernel is not only structural, since it ultimately relies on the semantic similarity of the primitive concepts on the leaves, assessed by comparing their extensions through a set kernel. Moreover, the kernel is applied to couples of individuals, after having lifted them to the concept level through realization operators (actually by means of approximations of the most specific concept, see [2]).

In [24], a set of kernels for individuals and for the various types of assertions in the ABox (on concepts, datatype properties, object properties) are presented. They should be composed for obtaining the final kernel; however, it is not really specified how such separate building blocks have to be integrated; the preliminary evaluation on specific classification problems regarded single kernels or simple additive combinations.

4.2. A Family of Epistemic Kernels for OWL

Language independent kernel functions for SW representations can be defined similarly to the metrics seen in the previous section. Differently from those defined in [23], these kernels are totally semantic and language independent. Jointly with a SVM, a kernel function will be used for learning classifiers that can be employed to perform inductive concept retrieval and query answering.

The family of kernels we propose, derived from the measure presented in §3.3, can be directly applied to individuals. It is parametrized on a set of features (concepts) that are used for its computation. Similarly to KFOIL [25], a sort of dynamic propositionalization takes place. However, in this setting the committee of concepts which are used as dimensions for the similarity function are not due to alternate versions of the same target concept but may vary freely, reflecting contextual knowledge. Also the form of the kernel function resembles that of the Minkowski’s metrics for vectors of numeric features:

**Definition 5. (DL-kernel functions)** Let $\mathcal{K} = \langle T, A \rangle$ be a knowledge base. Given a set of concept descriptions $F = \{F_1, F_2, \ldots, F_m\}$, a family of kernel functions

$$k_p^F : \text{Ind}(A) \times \text{Ind}(A) \mapsto [0, 1]$$

is defined as follows:
∀(a, b) ∈ Ind(\mathcal{A}) × Ind(\mathcal{A})

\begin{align*}
k_F^p(a, b) := \frac{1}{|F|} \left( \sum_{i=1}^{|F|} |\sigma_i(a, b)|^p \right)^{1/p}
\end{align*}

where \( p > 0 \) and \( \forall i \in \{1, \ldots, m\} \) the simple similarity function \( \sigma_i \) is defined:

\begin{align*}
\sigma_i(a, b) &= \begin{cases} 
1 & (\mathcal{K} \models F_i(a) \land \mathcal{K} \models F_i(b)) \lor (\mathcal{K} \models \neg F_i(a) \land \mathcal{K} \models \neg F_i(b)) \\
0 & (\mathcal{K} \models \neg F_i(a) \land \mathcal{K} \models F_i(b)) \lor (\mathcal{K} \models F_i(a) \land \mathcal{K} \models \neg F_i(b)) \\
1/2 & \text{otherwise}
\end{cases}
\end{align*}

The rationale for this kernel is that similarity between individuals is decomposed along with the similarity with respect to each concept in a given committee of features (concept definitions). Two individuals are maximally similar w.r.t. 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. Because of the OWA, sometimes a reasoner cannot assess the concept-membership, hence, since both possibilities are open, we assign an intermediate value to reflect such uncertainty. If the standard properties of similarity measures were required equivalence classes might be considered instead of mere individuals.

It is also worthwhile to note that this is indeed a family of kernels parametrized on the choice of the feature set. The effectiveness and also the efficiency of the measure computation strongly depends on the choice of the feature committee (feature selection). Optimal features can be learnt by the use of randomized optimization procedures \([14, 26]\).

Instance-checking is to be employed for assessing the value of the simple similarity functions \( \sigma_i \). Yet this is known to be computationally expensive (also depending on the specific DL language of choice). Alternatively, especially for largely populated ontologies which may be the objective of mining algorithms, a simple look-up may be sufficient.

The most important property of a kernel function is its validity (it must correspond to a dot product in a certain embedding space).

**Proposition 1. (validity)** Given an integer \( p > 0 \) and a committee of features \( \mathcal{F} \), the function \( k_F^p \) is a valid kernel.

This result can be assessed by proving the property by showing that the function can be obtained by composing simpler valid kernels through operations that guarantee the closure w.r.t. this property \([21]\). Specifically, since the similarity functions \( \sigma_i (i = 1, \ldots, n) \) correspond to matching kernels, the property follows from the closure w.r.t. power, sum, multiplication by a constant and kernel multiplication.

The intermediate value used in the uncertain cases (here 1/2) may be chosen more carefully so to reflect the inherent uncertainty related to the specific features.
Table 1. Facts concerning the ontologies employed in the experiments.

| Ontology   | language | #concepts | #obj. prop. | #data prop. | |Ind(\mathcal{A})|
|------------|----------|-----------|-------------|-------------|----------------|
| SWM        | $\text{ALCOF}(D)$ | 19        | 9           | 1           | 115           |
| BioPAX     | $\text{ALCHF}(D)$ | 28        | 19          | 30          | 323           |
| LUBM       | $\text{ALR}_1, \text{HI}(D)$ | 43        | 7           | 25          | 555           |
| NTN        | $\text{SHIF}(D)$ | 47        | 27          | 8           | 676           |
| SWSD       | $\text{AACH}$ | 258       | 25          | 0           | 732           |
| Financial  | $\text{ALCIF}$ | 60        | 17          | 0           | 1000          |

An alternative choice that is being experimented is related to the balance between the number of known individuals that belong to the feature concept and those that certainly belong to its negation.

5. Experimental Evaluation

The dissimilarity measure presented in Sect. 3 has been integrated in the NN procedure (see Sect. 3) while the DL-Kernel (Sect. 4.2) has been embedded in a SVM from the LIBSVM library\(^1\). Both methods have been tested by applying them to a number of retrieval and query answering problems. In the following, the results of the experiments for each method are reported.

In order to assess the validity of the algorithms presented in the previous sections, a number of OWL ontologies from different domains have been considered: \textsc{Surface-Water-Model} (SWM), \textsc{NewTestamentNames} (NTN) from the \textsc{Protege} library\(^m\), Semantic Web Service Discovery dataset\(^n\) (SWSD); an ontology generated by the Lehigh University Benchmark\(^o\) (LUBM); BioPax glycolysis ontology\(^p\) (BioPax) and \textsc{Financial} ontology\(^q\). Tab. 1 summarizes details concerning these ontologies.

The performance was evaluated comparing the responses given by the inductive classifiers to those returned by a standard reasoner\(^r\) as a baseline, and the following indices have been considered for the evaluation:

- \textit{match rate}: number of cases of individuals that got exactly the same classification by both classifiers with respect to the overall number of individuals;
- \textit{omission error rate}: amount of individuals for which inductive classifier could not determine whether they were relevant to the query or not (namely

\(^1\)Software downloadable at: http://www.csie.ntu.edu.tw/~cjlin/libsvm
\(^m\)http://protege.stanford.edu/plugins/owl/owl-library
\(^n\)https://www.uni-koblenz.de/IFB4/Institutes/IFI/AGStaab/Projects/xmedia/dl-tree.htm
\(^o\)http://swat.cse.lehigh.edu/projects/lubm/
\(^p\)http://www.biopax.org/Downloads/Level1v1.4/biopax-example-ecocyc-glycolysis.owl
\(^q\)http://www.cs.put.poznan.pl/alawrynowicz/financial.owl
\(^r\)We employed Pellet v. 1.5.1. See http://pellet.owldl.com
Table 2. Outcomes of the concept classification experiments with the NN procedure: averages ± standard deviations.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>match rate</th>
<th>commission error rate</th>
<th>omission error rate</th>
<th>induction rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWM</td>
<td>96.3 ± 02.6</td>
<td>00.0 ± 00.0</td>
<td>02.4 ± 01.6</td>
<td>01.3 ± 01.9</td>
</tr>
<tr>
<td>BioPAX</td>
<td>92.0 ± 11.7</td>
<td>07.5 ± 11.8</td>
<td>00.0 ± 00.1</td>
<td>00.5 ± 02.6</td>
</tr>
<tr>
<td>LUBM</td>
<td>98.3 ± 07.1</td>
<td>00.0 ± 00.0</td>
<td>01.3 ± 04.9</td>
<td>00.3 ± 02.6</td>
</tr>
<tr>
<td>NTN</td>
<td>91.0 ± 13.2</td>
<td>00.3 ± 01.3</td>
<td>05.0 ± 07.2</td>
<td>03.7 ± 10.7</td>
</tr>
<tr>
<td>SWSD</td>
<td>98.6 ± 04.4</td>
<td>00.0 ± 00.0</td>
<td>01.2 ± 03.4</td>
<td>00.2 ± 01.3</td>
</tr>
<tr>
<td>Financial</td>
<td>97.3 ± 06.8</td>
<td>02.4 ± 06.8</td>
<td>00.0 ± 00.1</td>
<td>00.3 ± 00.2</td>
</tr>
</tbody>
</table>

individuals classified as belonging to the class 0) while they were actually relevant (classified as +1 or −1 by the standard reasoner);

- **commission error rate**: amount of individuals found to be relevant to the query concept by the induced classifier, while they belong to its negation or vice-versa

- **induction rate**: amount of individuals found to be relevant to the query concept or to its negation, while either case is not logically derivable from the knowledge base

For each ontology, the concept classification task involved all the concepts occurring in the KB. Moreover, for the experiments on approximate query answering, 30 random queries were generated by composition of (2 through 8) primitive or defined concepts in each knowledge base by means of the operators of the related OWL sub-language. For all experiments, a 10-fold cross validation setting was adopted to get a good estimate of the performance.

5.1. **Experiments with the NN Procedure**

For the experiments with NN procedure, the simplest version of the distance ($d_F^1$) was employed with the committee of feature $F$ made by all concepts in the knowledge base (except the target one in case of the experiments on the classification task). The parameter $k$ was set to $\sqrt{|Ind(A)|}$, as advised in the instance-based learning literature. Yet we found experimentally that much smaller values could be chosen, resulting in the same classification.

Tab. 2 reports the outcomes of the classification experiments in terms of the evaluation indices. Preliminarily, note that the match rate was generally above 90%, a small amount of omission errors were made and some cases of induction were observed. As for the commission errors, the rate was quite low but in the cases of the BioPAX ontology that represent also a case where the highest variance was exhibited. On a careful review of the experiments, it was observed a few concepts in both ontologies turned out to be quite difficult to classify, namely the match rate in one case dropped to 50% in favor of the commission error rate, while scarce induction
Table 3. Outcomes of the query answering experiments with the NN procedure: averages ± standard deviations.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Match rate</th>
<th>Commission error rate</th>
<th>Omission error rate</th>
<th>Induction rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWM</td>
<td>93.3 ± 10.3</td>
<td>00.0 ± 00.0</td>
<td>02.5 ± 04.4</td>
<td>04.2 ± 10.5</td>
</tr>
<tr>
<td>BioPAX</td>
<td>99.9 ± 00.2</td>
<td>00.2 ± 00.2</td>
<td>00.8 ± 00.0</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>LUBM</td>
<td>99.2 ± 00.8</td>
<td>00.0 ± 00.0</td>
<td>00.8 ± 00.8</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>NTN</td>
<td>98.6 ± 01.5</td>
<td>00.0 ± 00.1</td>
<td>00.8 ± 01.1</td>
<td>00.0 ± 01.4</td>
</tr>
<tr>
<td>SWSD</td>
<td>97.5 ± 03.7</td>
<td>00.0 ± 00.0</td>
<td>01.8 ± 02.6</td>
<td>00.8 ± 01.5</td>
</tr>
<tr>
<td>Financial</td>
<td>99.5 ± 00.8</td>
<td>00.3 ± 00.7</td>
<td>00.0 ± 00.0</td>
<td>00.2 ± 00.2</td>
</tr>
</tbody>
</table>

or omission errors were observed. Particularly, for the BioPAX, more information on the concept disjointness and the limited amount of positive instances available for these hard concepts more easily determined the cases of commission errors. Also in the experiments with NTN a higher variance was observed, but the less cases of match resulted in an increase of the cases of induction and, limitedly, of the omission errors.

Tab. 3 reports the outcomes of the query answering experiments in terms of the evaluation indices. Preliminarily, it is important to note that, in each experiment, the commission error was low or absent. This means that the search procedure is quite accurate: it did not make critical mistakes i.e. cases when an individual is deemed as an instance of a concept while it really is not (it belongs to its negation). Also the omission error rate is quite low, yet it is more typical over the ontologies that were considered. A noteworthy difference was observed for the case of the SWS knowledge base for which we found the lowest match rate and the highest variability in the results over the various concepts. Of course these results are influenced by the particular concepts that were randomly generated as a number of 30 is very limited compared to the large number of queries that can be composed using concepts and roles defined in each ontology.

If we compare these outcomes with those reported in other works on similar tasks [13], where the highest average match rate observed was about 80%, we find a significant increase of the performance due to the accuracy of the new measure. Also the elapsed time (not reported here) was much less because of the different dissimilarity measure: once the values for the dissimilarity functions are pre-computed, the efficiency of the classification, which depends a lot on the computation of the dissimilarity, gains a lot of speed-up.

The usage of all concepts in each ontology for the set of features $F$ made the measure quite accurate, which is the reason why the procedure resulted quite conservative as regards inducing new assertions. In many cases, it matched rather faithfully the reasoner decisions, even when its response was an unknown class-membership. The cases of induction are more interesting because they suggest new assertions
Table 4. Results (average and standard deviation) of the experiments on concept classification with LIBSVM integrating the DL-kernel.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>match rate</th>
<th>commission error rate</th>
<th>omission error rate</th>
<th>induction rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWM</td>
<td>86.2 ± 17.6</td>
<td>08.0 ± 16.1</td>
<td>05.8 ± 04.6</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>BioPax</td>
<td>92.1 ± 11.3</td>
<td>00.6 ± 02.7</td>
<td>00.0 ± 00.0</td>
<td>07.4 ± 11.4</td>
</tr>
<tr>
<td>LUBM</td>
<td>93.0 ± 12.1</td>
<td>03.5 ± 12.2</td>
<td>03.5 ± 04.9</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>NTN</td>
<td>91.0 ± 13.1</td>
<td>04.0 ± 11.6</td>
<td>04.8 ± 07.5</td>
<td>00.3 ± 01.4</td>
</tr>
<tr>
<td>SWSD</td>
<td>98.7 ± 04.1</td>
<td>00.0 ± 00.0</td>
<td>01.3 ± 04.1</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>FINANCIAL</td>
<td>97.2 ± 06.8</td>
<td>00.3 ± 00.2</td>
<td>00.0 ± 00.1</td>
<td>02.4 ± 06.8</td>
</tr>
</tbody>
</table>

which cannot be logically derived by using a deductive reasoner, and they might be used to complete a knowledge base [9], e.g. after being validated by an ontology engineer and/or a domain expert.

5.2. Experimental Evaluation of the Epistemic Kernel

In order to experimentally assess the effectiveness of the other non-parametric method embedding the epistemic kernel in a SVM, the instance classification task has been performed inducing this type of classifiers on the same ontologies (detailed in Tab. 1).

Again, the inductive method adopting a classifier induced from a set of training individuals through the SVM embedding a DL-Kernel (p = 2) was applied to classify all the individuals in each ontology. Specifically, for each ontology, after inducing the classifier from a set of training instances, this served to perform instance-checking w.r.t. the test concepts (defined ones or randomly generated queries).

The performance of the classifier induced by the SVM was evaluated by comparing its responses to those returned by the reasoner used as baseline in the previous experiments. The experiment has been performed by adopting the ten-fold cross validation procedure.

A similar experimental setting has been considered in [24] with an exemplified version of the GALEN Upper Ontology. There, the ontologies have been randomly populated and only seven concepts have been considered while no roles have been taken into account. Differently from this case, we did not apply any changes on the considered ontologies. Moreover, for assessing the performance of the method the standard IR indices of precision, recall and F-measure have been used.

The average measures obtained over all the concepts in each ontology are re-

---

Footnotes:
- The feature set \( F \) for computing the epistemic kernel was made by all concepts in the considered ontology.
- http://www.cs.man.ac.uk/~rector/ontologies/simple-top-bio
- Due to the lack of information for replicating the ontology used in [24], a comparative experiment with the proposed kernel framework cannot be performed.
ported in Tab. 4, jointly with their standard deviation. By looking at the table, it is important to note that, for every ontology, the commission error is almost null. This means that the classifier did not make critical mistakes, i.e. cases when an individual is deemed as an instance of a concept while it really is known to be an instance of another disjoint concept. In the same time it is important to note that a high match rate is registered for every ontology. The most difficult cases are those (like the SWM ontology) were the limited number of individuals in the ontology caused the availability of few examples or counterexamples for building the inductive model. Indeed, by considering again also Tab. 1, it is interesting to observe that the match rate increases with the increase of the complexity of the considered ontology. This is because the performance of a statistical method improves with the augmentation of the set of the available individuals, that means that there is more information for inducing classifiers that can separate positive from negative examples. Almost always the SVM-based classifier is able to induce new knowledge. However, a more conservative behavior w.r.t. the previous experiment with the \textit{kNN} method has been also registered, indeed the omission error rate is not null (even though it is very close to 0). To decrease the tendency to such a conservative behavior of the method, a threshold could be introduced so to avoid returning the unknown class-membership decision.

The other experiment regarded testing the SVM-based method when performing inductive concept retrieval w.r.t. new query concepts built from the considered ontology. The method has been applied to perform a number of retrieval problems applied to the considered ontologies again using the SVM integrating a \textit{DL-kernel} function. A number of queries were randomly generated by applying the available constructors to primitive and/or defined concepts and roles from each ontology. The generated concepts were checked not only for satisfiability but also for having both positive and negative instances explicitly in the ontology. Again, a ten-fold cross validation was performed for each dataset.

The outcomes of this experiment are reported in Tab. 5, from which it is possible to observe that the behavior of the classifier on these concepts is not very dissimilar with respect to the outcomes of the previous experiment (see Tab. 4). These queries were expected to be harder than the previous ones which correspond to the very primitive or defined concepts for the various ontologies. Specifically, the commission error rate was low for all but two ontologies (BioPAX and LUBM) for which some very difficult queries were generated which raised this rate beyond 10% and consequently also the standard deviation values.

By comparing these results with those obtained by the use of the NN approach (see Tab. 2 and 3) it is possible to assert that both methods showed high accuracy in performing the classification task, even if the NN method has resulted to be more accurate when a lower number of instances are available for performing the learning task. Besides the LIBSVM offers a number of parameters that could have been adjusted for each learning problem in order to obtain a better performance. We decided to leave the defaults unchanged so to allow for a comparison with further
Table 5. Results (average and standard deviation) of the experiments for performing query answering with LIBSVM integrating the DL-kernel.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>match</th>
<th>induction</th>
<th>omission</th>
<th>commission</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWM</td>
<td>82.3 ± 21.8</td>
<td>09.1 ± 16.5</td>
<td>08.6 ± 08.5</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>BioPAX</td>
<td>84.0 ± 14.6</td>
<td>00.0 ± 00.0</td>
<td>00.0 ± 00.0</td>
<td>16.0 ± 15.6</td>
</tr>
<tr>
<td>LUBM</td>
<td>76.8 ± 19.7</td>
<td>05.8 ± 05.9</td>
<td>00.0 ± 00.0</td>
<td>17.5 ± 20.9</td>
</tr>
<tr>
<td>NTN</td>
<td>80.4 ± 17.0</td>
<td>08.2 ± 16.9</td>
<td>10.0 ± 10.1</td>
<td>01.4 ± 03.0</td>
</tr>
<tr>
<td>SWSD</td>
<td>97.9 ± 03.8</td>
<td>00.0 ± 00.0</td>
<td>02.1 ± 03.8</td>
<td>00.0 ± 00.0</td>
</tr>
<tr>
<td>FINANCIAL</td>
<td>97.9 ± 03.4</td>
<td>00.4 ± 00.2</td>
<td>00.0 ± 00.1</td>
<td>01.7 ± 03.4</td>
</tr>
</tbody>
</table>

6. Conclusions

We investigated on the application of non-parametric statistical multi-relational learning methods in the context of the SW representations in order to offer alternative ways for performing classification in ontologies that may overcome the limitations of the standard (purely logic) methods. Specifically, a family of semi-distance and kernel functions have been defined for OWL descriptions. They have been integrated respectively in a NN classification procedure and a SVM for inducing a statistical classifier working with the complex representations. The resulting classifiers were tested on inductive retrieval and classification problems on real ontologies from standard repositories.

The peculiarity of learning from ontologies that naturally assume an open-world semantics required a specific assessment of the performance of the induced classifiers, since conclusions have to deal with the chance of uncertainty: some instances cannot be attributed to a class or to its negation. Hence, we have proposed four measures for evaluating the outcomes of an induced classifier reflecting the various cases of alignment between the decision made by the classifier (reasoner) w.r.t. the three possible values.

Experimentally, it was shown that performance of both classifiers are not only comparable to a standard deductive reasoner, but they are also able to induce new knowledge, which is not logically derivable. Particularly, an increase in prediction accuracy was observed for ontologies that are homogeneously populated (similar to a database).

Adopting a two-way decision procedure based on the likelihood of the response determined by the inductive classifier in principle the method could possibly go beyond the mere classification/retrieval task and offer a chance for getting more insight in the KB and even rank the hits. Even more so, the induced classification results can be exploited for predicting or suggesting missing information about individuals, thus completing large ontologies. Specifically, it can be used to semi-automatize the population of an ABox. Indeed, the new assertions can be suggested
to the knowledge engineer that has only to validate their inclusion. This constitutes a new approach in the SW context, since the efficiency of the statistical and numerical approaches and the effectiveness of a symbolic representation have been combined.

References


