

# REDUCE: A Reduced Coulomb Energy Network Method for Approximate Classification

Nicola Fanizzi, Claudia d'Amato, and Floriana Esposito

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

**Abstract.** In order to overcome the limitations of purely deductive approaches to the tasks of classification and retrieval from ontologies, inductive (instance-based) methods have been proposed as efficient and noise-tolerant alternative. In this paper we propose an original method based on non-parametric learning: the REDUCED COULOMB ENERGY (RCE) Network. The method requires a limited training effort but it turns out to be very effective during the classification phase. Casting retrieval as the problem of assessing the class-membership of individuals w.r.t. the query concepts, we propose an extension of a classification algorithm using RCE networks based on an *entropic* similarity measure for OWL. Experimentally we show that the performance of the resulting inductive classifier is comparable with the one of a standard reasoner and often more efficient than with other inductive approaches. Moreover, we show that new knowledge (not logically derivable) is induced and the likelihood of the answers may be provided.

## 1 Introduction

The inherent incompleteness and accidental inconsistency of knowledge bases (KBs) in the Semantic Web (SW) requires approximate query answering methods for retrieving resources. This activity is generally performed by means of logical approaches that try to cope with the problems mentioned above. This has given rise to alternative methods [15, 10, 9, 13, 11, 14].

Inductive methods applied to specific tasks are known to be often more efficient and noise-tolerant. Lately, first steps have been taken to apply classic machine learning techniques for building inductive classifiers to the complex representation and semantics adopted in the context of the SW [7], especially through non-parametric methods [4, 8]. Instance-based inductive methods may help a knowledge engineer populate ontologies [2]. Some methods are also able to complete ontologies with probabilistic assertions, allowing for sophisticated approaches to dealing with uncertainty [12].

In this paper we propose a novel method for inducing a classifier that can have several applications. Mainly it may naturally be employed as an alternative method for performing *concept retrieval* [1]. Even more so, like its predecessors mentioned above, it is able to determine a likelihood measure of the

induced class-membership assertions which is important for *approximate query answering*. Some assertions could not be logically derived, but may be highly probable according to the classifier; this may help to cope with the uncertainty caused by the inherent incompleteness of the KBs even in absence of an explicit probabilistic model.

Specifically, we propose to answer queries adopting an *instance-based* classifier induced by a non-parametric learning method, the REDUCED COULOMB ENERGY (RCE) network [5], that has been extended to be applied to the standard representations of the SW via a semantic similarity measure for individual resources. As with other similarity-based methods, a retrieval procedure may seek for individuals belonging to query concepts, exploiting the analogy with other training instances, namely the classification of the nearest ones (w.r.t. the similarity measure).

Differently from lazy-learning approaches experimented in the past [4], that do not require training, yet more similarly to the methods based on *kernel machines* [3, 8], the new method is organized in two phases: 1) in the *training* phase a simple network based on prototypical individuals (parametrized for each prototype) is trained to detect the membership of further individuals w.r.t. some query concept; 2) this network is then exploited by the classifier, during the *classification* phase, to make a decision on the class-membership of an individual w.r.t. the query concept on the grounds of a likelihood estimate. The network parameters correspond to the radii of hyperspheres centred at the training individuals in the context of the metric space determined by some measure. The classification of an individual is induced observing the balls it belongs to and on its distance from their centers (training prototypes). The efficiency of the method derives from limiting the expensive model construction to a small set of training individuals, while the resulting RCE network can be exploited in the next phase to efficiently classify a large number of individuals.

A similarity measure derived from a family of semantic pseudo-metrics [4] was exploited. This language-independent measure assesses the similarity of two individuals by comparing them on the grounds of their behavior w.r.t. a committee of features (concepts), namely (a subset of) those defined in the KB which can be thought as a context; alternatively, it may be generated to this purpose through randomized search algorithms aiming at optimal committees to be run in advance [6]. Besides, since different features may exhibit a different relevance w.r.t. the similarity value to be determined, each feature is weighted on the grounds of the *quantity of information* it conveys. This weight is determined by an *entropic* measure. The rationale is that the more general a feature (or its negation) the less likely is should be used for distinguishing the individuals.

The resulting system REDUCE allowed for an intensive experimentation of the method on performing approximate query answering with a number ontologies from public repositories. Like in previous works [4, 8], the predictions were compared to assertions that were logically derived by a deductive reasoner. The experiments showed that the classification results are comparable (although

slightly less complete) and also that the classifier is able to induce new knowledge that is not logically derivable.

The paper is organized as follows. The basics of the instance-based approach applied to the standard representations are recalled in Sect. 2. The next Sect. 3 presents the semantic similarity measures adopted in the retrieval procedure. Sect. 4 reports the outcomes of the experiments performed with the implementation of the procedure. Possible developments are finally examined in Sect. 5.

## 2 RCE Network Training and Classification

In the following sections, we assume that concept descriptions are defined in terms of a generic language based on OWL-DL that may be mapped to *Description Logics* (DLs) with the standard model-theoretic semantics (see the handbook [1] for a thorough reference).

A *knowledge base*  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  contains a *TBox*  $\mathcal{T}$  and an *ABox*  $\mathcal{A}$ .  $\mathcal{T}$  is a set of axioms that define concepts (and relationships).  $\mathcal{A}$  contains factual assertions concerning the individuals (resources). The set of the individuals occurring in  $\mathcal{A}$  is denoted with  $\text{Ind}(\mathcal{A})$ . The *unique names assumption* may be made on the such individuals, that are represented in OWL by their URIs.

As regards the required inference services, like all other instance-based methods, the inductive procedure requires performing *instance-checking* [1], which roughly amounts to determining whether an individual, say  $a$ , belongs to a concept extension, i.e. whether  $C(a)$  holds for a certain concept  $C$ . This is denoted by  $\mathcal{K} \models C(a)$ . This service is generally provided by a reasoner. Note that because of the open-world semantics, it may be unable to give a positive or negative answer to a class-membership query.

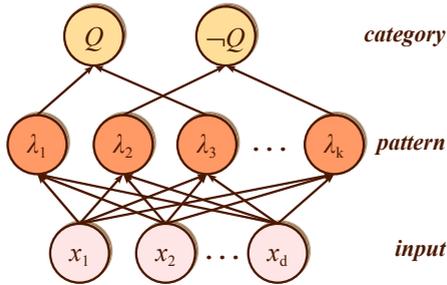
### 2.1 Basics

An approximate classification method is proposed for individuals in the context of semantic knowledge bases. The method requires training an RCE network, which can be successively exploited for an inductive classification procedure. Moreover, a likelihood measure of the decisions made by the inductive procedure can be provided.

In *instance-based* learning [5, 16], the basic mechanism determines the membership of a query instance to some concept in accordance with the membership of the most similar instance(s) with respect to a similarity measure.

This may be formalized as the induction of an estimate for a discrete-valued target hypothesis function  $h : \text{TrSet} \rightarrow V$  from a space of instances  $\text{TrSet}$  to a set of values  $V = \{v_1, \dots, v_s\}$ , standing for alternative classes to be predicted. In the specific case of interest [4], the adopted value set is  $V = \{+1, -1, 0\}$ , where the three values denote, respectively, membership, non-membership, and uncertainty. The values of  $h$  for the training instances are determined as follows:

$$\forall x_i \in \text{TrSet}: \quad h_Q(x_i) = \begin{cases} +1 & \mathcal{K} \models Q(x_i) \\ -1 & \mathcal{K} \models \neg Q(x_i) \\ 0 & \text{otherwise} \end{cases}$$



**Fig. 1.** The structure of a Reduced Coulomb Energy network

Note that normally  $|TrSet|$  is much less than  $|\text{Ind}(\mathcal{A})|$  i.e. only a limited number of training instances (exemplars) is needed, especially if they can be carefully selected among the prototypical ones for the various regions of the search space.

Let  $x_q$  be the query instance whose class-membership is to be determined. Given a dissimilarity measure  $d$ , in the  $k$ -Nearest Neighbor method ( $k$ -NN) [4] the estimate of the hypothesis function for the query individual may be determined by:  $\hat{h}_Q(x_q) = \text{argmax}_{v \in V} \sum_{i=1}^k \delta(v, h_Q(x_i)) / d(x_i, x_q)^2$  where  $\delta$  returns 1 in case of matching arguments and 0 otherwise.

## 2.2 Training

The RCE method<sup>1</sup> adopts a similar non-parametric approach. The construction of the RCE model can be implemented as training a (sort of neural) network whose structure, for the binary problem of interest, is sketched in Fig. 1.

The input layer receives the information from the individuals. The middle layer of patterns represents the features that are constructed for the classification; each node in this layer is endowed with a parameter  $\lambda_j$  representing the radius of the hypersphere centered in the  $j$ -th prototype individual of the training set. During the training phase, each coefficient  $\lambda_j$  is adjusted so that the spherical region is as large as possible, provided that no training individual of a different class is contained. Each pattern node is connected to one of the output nodes representing the predicted class. In our case we have only two categories representing membership and non-membership w.r.t. the query concept  $Q$ .

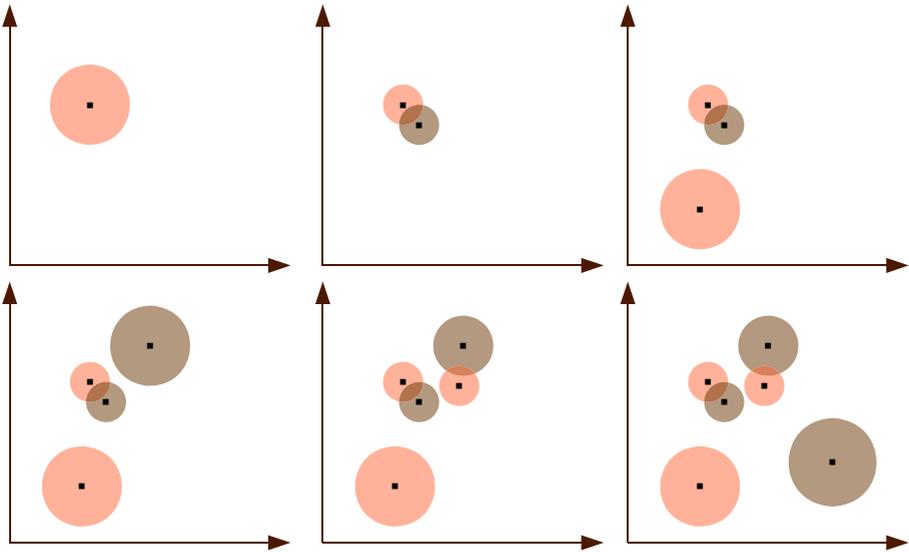
The training algorithm is sketched in Fig. 2. Suppose the training instances (a.k.a. examples) are labeled with their correct classification  $\langle x_i, h_Q(x_i) \rangle$ , where  $h_Q(x_i) \in V$ , as seen before. In this phase, each parameter  $\lambda_j$ , that represents the radius of a hypothetical hypersphere centered at the input example, is adjusted to be as large as possible (they are initialized with a maximum radius), provided that the resulting region does not enclose counterexamples. As new individuals are processed, each such radius  $\lambda_j$  is decreased accordingly (and can never increase). In this way, each pattern unit can enclose several prototypes, but only those having the same category label.

<sup>1</sup> Borrowing its name from electrostatics, where energy is associated with charged particles [5].

**input**  
 $TrSet = \{ \langle x_i, h_Q(x_i) \rangle \}$ : set of training examples  
**output**  
 $w_{jk}, \lambda_j, a_{cj}$ : RCE Network weights

1. **begin**
2. initialize  $\epsilon \leftarrow$  small parameter;  $\lambda_{\max} \leftarrow$  max radius
3. **for**  $j \leftarrow 1$  **to**  $|TrSet|$  **do**
  - (a) train weight:  $w_{jk} \leftarrow x_k$
  - (b) find nearest counterexample:  $\bar{x} \leftarrow \arg \min_{x \in \mathcal{C}_j} d(x, x_j)$   
 where  $\mathcal{C}_j = \{ x \in TrSet \mid h_Q(x_j) \neq h_Q(x) \}$
  - (c) set radius:  $\lambda_j \leftarrow \min[\max(d(\bar{x}, x_j), \epsilon), \lambda_{\max}]$
  - (d) **if**  $(h_Q(x_j) = +1)$  **then**  $a_{Qj} \leftarrow 1$  **else**  $a_{-Qj} \leftarrow 1$
4. **end**

**Fig. 2.** RCE training algorithm



**Fig. 3.** Evolution of the model built by a RCE network: the centers of the hyperspheres represent the prototypical individuals

Fig. 3 shows the evolution of the model (two-colored decision regions for binary problems) that becomes more and more complex as long as training instances are processed. Different colors represent the different membership predictions. Uncertain regions present overlapping colors.

It is easy to see that the complexity of the method, when no particular optimization is made, is  $O(n^2)$  in the number of training examples ( $n = |TrSet|$ ) as the weights are to be adjusted for all of them and each modification of the radius  $\lambda_j$  requires a search for the counterexample with minimal distance among the other training instances.

The method can be used both in *batch* and *on-line* mode. The latter incremental model is particularly appealing when the application may require performing intensive queries on the same query concept and new instances are likely to be made available along the time.

There are several subtleties that may be considered. For instance, if the radius of a pattern unit becomes too small (i.e., less than some threshold  $\lambda_{\min}$ ), then it indicates highly overlapping different categories. In that case, the pattern unit is called a *probabilistic* unit, and so marked.

The method is related to other non-parametric methods, *k-NN estimation* and *Parzen windows* [5]. In particular the Parzen windows method uses fixed window sizes that could lead to some difficulties: in some regions a small window width is appropriate while elsewhere a large one would be better. The *k-NN* method uses variable window sizes increasing the size until enough samples are enclosed. This may lead to unnaturally large windows when sparsely populated regions are targeted. In the RCE method, the window sizes are adjusted until points of a different category are encountered.

### 2.3 Classification

The classification of a query individual  $x_q$  using the trained RCE network is quite simple in principle. As shown in the basic (*vanilla*) form of the classification algorithm depicted in Fig. 4, the set  $N(x_q) \subseteq TrSet$  of the nearest training instances is built on the grounds of the hyperspheres (determined by the  $\lambda_j$  weights) the query instance belongs to. Each hypersphere has a related classification determined by the prototype at its center (see Fig. 5). If all prototypes agree on the classification this value is returned as the induced estimate, otherwise the query individual is deemed as ambiguous w.r.t.  $Q$ , which represents the default case.

#### input

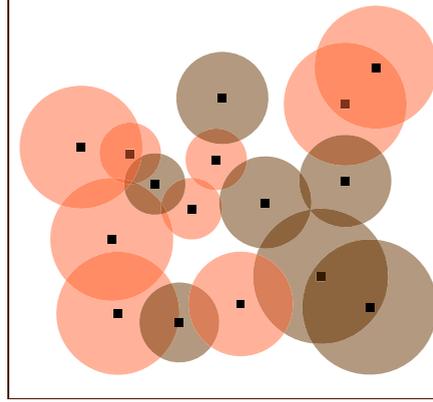
$x_q$ : query individual  
 $TrSet$ : set of training examples  
 $\lambda_j$ : parameters of the trained RCE network

#### output

$\hat{h}_Q(x_q)$ : estimated classification

1. **begin**
2. initialize  $k \leftarrow 0$ ;  $N(x_q) \leftarrow \emptyset$
3. **for**  $j \leftarrow 1$  to  $|TrSet|$  **do**
  - **if**  $d(x_q, x_j) < \lambda_j$ 
    - then**  $N(x_q) \leftarrow N(x_q) \cup \{x_j\}$
4. **if**  $(\forall x, x' \in N(x_q) : h_Q(x) = h_Q(x'))$  all share the same class
  - then return**  $h_Q(x)$ , shared class of all  $x \in N_{set}$
  - else return** 0 // *uncertain case*
5. **end**

**Fig. 4.** Vanilla RCE classification



**Fig. 5.** A representation of the model built by an RCE network used for classification: regions with different colors represent different classifications for instances therein. The areas with overlapping colors or outside of the scope of the hyperspheres represent regions of uncertain classification (see the proposed enhancement of the procedure in this case).

It is easy to see that this inductive classification procedure is linear in the number of training instances and it could be optimized by employing specific data structures, such as *kD-tress* or *ball-trees* [16], which would allow a faster search of the including hyperspheres and their centers.

In case of uncertainty, this procedure may be enhanced in the spirit of *k*-NN classification recalled above. If a query individual is located in more than one hypersphere, instead of a catch-all decision like the one made at step 4. of the algorithm, requiring all involved prototypes to agree on their classification in a sort of *voting* procedure, each vote may be weighted by the similarity of the query individual w.r.t. the hypersphere center in terms of a similarity measure<sup>2</sup>  $s$ , and the classification should be defined by the class whose prototypes are globally the closest, considering the difference between the closeness values of the query individual from the centers classified by either class. Indeed, one may also consider the *signed* vote, where the sign is determined by the classification of each selected training prototype, and sum up these votes determining the classification with a sign function.

Formally, suppose the nearest prototype set  $N(x_q)$  has been determined. The decision function is defined:

$$g(x_q) = \sum_{x_j \in N(x_q)} h_Q(x_j) \cdot s(x_j, x_q)$$

Then step 4. in the procedure becomes:

4. **if** ( $|g(x_q)| > \theta$ ) **then return**  $\text{sgn}(g(x_q))$  **else return** 0

<sup>2</sup> It may be derived from a distance or dissimilarity measure by means of simple transformations, as shown later.

where  $\theta \in ]0, 1]$  is a tolerance threshold for the uncertain classification (*uncertainty threshold*). We may foresee that higher values of this threshold make the classifier more *skeptical* in uncertain cases, while lower values make it more credulous in suggesting  $\pm 1$ .

#### 2.4 Likelihood of the Inductive Classification

The analogical 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 inductive procedure, one may resort to an approach that is similar to the one applied with the  $k$ NN procedure [4]. It is convenient to decompose the decision function  $g(x)$  in three components corresponding to the values  $v \in V : g_v(x)$  and use those weighted votes. Specifically, given the nearest training individuals in  $N(x_q) = \{x_1, \dots, x_k\}$ , the values of the decision function should be normalized as follows, producing a likelihood measure:

$$\ell(\hat{h}(x_q) = v \mid N(x_q)) = \frac{g_v(x_q)}{\sum_{u \in V} g_u(x_q)}$$

which can be written also:

$$\ell(\hat{h}(x_q) = v \mid N(x_q)) = \frac{\sum_{j=1}^k \delta(v, h_Q(x_j)) \cdot s(x_q, x_j)}{\sum_{u \in V} \sum_{h=1}^k \delta(u, h_Q(x_h)) \cdot s(x_q, x_h)}$$

For instance, the likelihood of the assertion  $Q(x_q)$  corresponds to the case when  $v = +1$  (i.e. to  $g_1(x_q)$ ). This could be used in case the application requires that the hits be ranked along with their likelihood values.

### 3 Similarity Measures

The method described in the previous section relies on a notion of similarity which should be measured by means of specific metrics which are to be sensible to the similarity/difference between the individuals. To this purpose various definitions have been proposed [6, 4, 8].

Given a dissimilarity measure  $d$  with values in  $[0, 1]$  belonging to the family of pseudo-metrics defined in [4], the easiest way to derive a similarity measure would be:  $\forall a, b: s(a, b) = 1 - d(a, b)$  or  $s(a, b) = 1/d(a, b)$ . The latter case needs some correction to avoid the undefined cases.

A more direct way follows the same ideas that inspired the mentioned metrics. Indeed, these measures are based on the idea of comparing the semantics of the input individuals along a number of dimensions represented by a committee of concept descriptions. Indeed, on a semantic level, similar individuals should behave similarly with respect to the same concepts.

More formally, the individuals are compared on the grounds of their semantics w.r.t. a collection of concept descriptions, say  $F = \{F_1, F_2, \dots, F_m\}$ , which

stands as a group of discriminating *features* expressed in the language taken into account. In its simple formulation, a family of similarity functions for individuals inspired to Minkowski's norms can be defined as follows [8]:

**Definition 3.1 (family of similarity measures).** *Let  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  be a knowledge base. Given a set of concept descriptions  $\mathbf{F} = \{F_i\}_{i=1}^m$  and a normalized vector of weights  $\mathbf{w} = (w_1, \dots, w_m)^t$ , a family of similarity functions*

$$s_p^{\mathbf{F}} : \text{Ind}(\mathcal{A}) \times \text{Ind}(\mathcal{A}) \rightarrow [0, 1]$$

is defined as follows:

$$\forall a, b \in \text{Ind}(\mathcal{A})$$

$$s_p^{\mathbf{F}}(a, b) = \frac{1}{m} \left[ \sum_{i=1}^m w_i |\sigma_i(a, b)|^p \right]^{1/p}$$

where  $p > 0$  and  $\forall i \in \{1, \dots, m\}$  the similarity function  $\sigma_i$  is defined by:

$$\forall a, b \in \text{Ind}(\mathcal{A})$$

$$\sigma_i(a, b) = \begin{cases} 0 & \text{if } [\mathcal{K} \models F_i(a) \text{ and } \mathcal{K} \models \neg F_i(b)] \text{ or } [\mathcal{K} \models \neg F_i(a) \text{ and } \mathcal{K} \models F_i(b)] \\ 1 & \text{if } [\mathcal{K} \models F_i(a) \text{ and } \mathcal{K} \models F_i(b)] \text{ or } [\mathcal{K} \models \neg F_i(a) \text{ and } \mathcal{K} \models \neg F_i(b)] \\ \frac{1}{2} & \text{otherwise} \end{cases}$$

The rationale of the measure is summing the partial similarity w.r.t. the single concepts. Functions  $\sigma_i$  assign the weighted maximal similarity to the case when the individuals exhibit the same behavior w.r.t. the given feature  $F_i$ , and null similarity when they belong to disjoint features. An intermediate value is assigned to the case when reasoning cannot ascertain one of such required class-memberships.

As regards the vector of weights  $\mathbf{w}$  employed in the family of measures, they should reflect the impact of the single feature concept w.r.t. the overall similarity. As mentioned, this can be determined by the quantity of information conveyed by a feature, which can be measured as its entropy. Namely, the extension of a feature  $F_i$  w.r.t. the whole domain of objects may be probabilistically quantified as  $P_i^+ = |F_i^{\mathcal{I}}|/|\Delta^{\mathcal{I}}|$  (w.r.t. the canonical interpretation  $\mathcal{I}$  whose domain is made up by the very individual names occurring in the ABox [1]). This can be roughly approximated with:  $P_i^+ = |\text{retrieval}(F_i)|/|\text{Ind}(\mathcal{A})|$ . Hence, considering also the probability  $P_i^-$  related to its negation and that related to the unclassified individuals (w.r.t.  $F_i$ ),  $P_i^U = 1 - (P_i^+ + P_i^-)$ , one may determine an entropic measure for the discernibility yielded by the feature:

$$h_i = - [P_i \log(P_i) + P_i^- \log(P_i^-) + P_i^U \log(P_i^U)]$$

These measures may be normalized for providing a good set of weights for distance or similarity measures  $w_i = h_i/||\mathbf{h}||$ .

## 4 Experimentation

The REDUCE system implements the training and classification procedures explained in the previous sections, borrowing the simplest metrics of the family (i.e., with  $p = 1$ ), for the sake of efficiency.

**Table 1.** Facts concerning the ontologies employed in the experiments

ontology	DL language	#concepts	#object prop.	#data prop.	#individuals
SWM	$\mathcal{ALCOF}(\mathcal{D})$	19	9	1	115
BioPAX	$\mathcal{ALCHF}(\mathcal{D})$	28	19	30	323
LUBM	$\mathcal{ALR}^+HI(\mathcal{D})$	43	7	25	555
NTN	$\mathcal{SHLF}(\mathcal{D})$	47	27	8	676
SWSD	$\mathcal{ALCH}$	258	25	0	732
FINANCIAL	$\mathcal{ALCIF}$	60	17	0	1000

Its performance has been tested in a number of classification problems that had been previously approached with other inductive methods [4, 8]. This allows the comparison of the new system to other inductive methods.

#### 4.1 Experimental Setting

A number of ontologies from different domains represented in OWL have been selected, namely: SURFACE-WATER-MODEL (SWM), NEWTESTAMENTNAMES (NTN) from the Protégé library<sup>3</sup>, the Semantic Web Service Discovery dataset<sup>4</sup> (SWSD), an ontology generated by the Lehigh University Benchmark<sup>5</sup> (LUBM), the BioPax glycolysis ontology<sup>6</sup> (BioPax) and the FINANCIAL ontology<sup>7</sup>. Table 1 summarizes details concerning these ontologies.

For each ontology, 100 satisfiable query concepts were randomly generated by composition (conjunction and/or disjunction) of (2 through 8) primitive and defined concepts in each knowledge base. The query concepts were also guaranteed to have instances in the ABox. Query concepts were constructed so to be endowed with at least a 50% of the ABox individuals classified positive and negative instances. The performance of the inductive method was evaluated by comparing its responses to those returned by a standard reasoner<sup>8</sup> as a baseline.

Experimentally, it was observed that large training sets make the similarity measures (and consequently the inductive procedure) very accurate. The simplest similarity measure ( $s_1^F$ ) was employed from the *entropic* family, using all the named concepts in the knowledge base for determining the committee of features  $F$  with no further optimization.

In order to lower the variance due to the composition of the specific training/test sets during the various runs, for each choice of query concepts, the experiments were replicated (250 folds) and the rates averaged according to the standard *.632+ bootstrap* procedure [5], which is based on sampling with replacement,

<sup>3</sup> <http://protege.stanford.edu/plugins/owl/owl-library>

<sup>4</sup> <https://www.uni-koblenz.de/FB4/Institutes/IFI/AGStaab/Projects/xmedia/dl-tree.htm>

<sup>5</sup> <http://swat.cse.lehigh.edu/projects/lubm>

<sup>6</sup> <http://www.biopax.org/Downloads/Level1v1.4/biopax-example-ecocyc-glycolysis.owl>

<sup>7</sup> <http://www.cs.put.poznan.pl/alawrynowicz/financial.owl>

<sup>8</sup> PELLET v. 2.0.0rc3 was employed.

hence producing tests sets of, approximately, one third of the individuals occurring in the ABoxes.

## 4.2 Results

In order to be able to compare the outcomes with those of previous experiments, we adopted the same evaluation indices which are briefly recalled here:

- *match rate*: rate of individuals that were classified with the same value ( $v \in V$ ) by both the inductive and the deductive classifier;
- *omission error rate*: rate of individuals for which the inductive method could not determine whether they were relevant to the query or not while they were actually relevant according to the reasoner (0 vs.  $\pm 1$ );
- *commission error rate*: rate of individuals that were classified as belonging to the query concept, while they belong to its negation or vice-versa ( $+1$  vs.  $-1$  or  $-1$  vs.  $+1$ );
- *induction rate*: rate of individuals inductively classified as belonging to the query concept or to its negation, while either case is not logically derivable from the knowledge base ( $\pm 1$  vs. 0);

We report in the following the outcomes of three different sessions with different choices of the parameters.

**First Session.** Table 2 reports the outcomes in terms of the indices in an experimentation where the uncertainty threshold was set to 0.3 and  $\epsilon = .01$ , which corresponds to a propensity for credulous classification.

Preliminarily, it is important to note that, in each experiment, the commission error was quite low. This means that the inductive search procedure is quite accurate, namely it did not make critical mistakes attributing an individual to a concept that is disjoint with the right one. The most difficult ontologies in this respect are those that contain many disjointness axioms, thus making more unlikely to have individuals with an unknown classification. Also the omission error rate was generally quite low, yet more frequent than the previous type of error.

It is noticeable also the induction rate which is generally quite high since a low uncertainty threshold makes the inductive procedure more prone to give a positive/negative classification in case a query individual is located in more than one hypersphere labeled with different classes.

From the instance retrieval point of view, the cases of induction are interesting because they suggest new assertions which cannot be logically derived by using a deductive reasoner yet they might be used to complete a knowledge base [2], e.g. after being validated by an ontology engineer. For each new candidate assertion, the estimated likelihood measure may be employed to assess its probability and hence decide on its inclusion in the KB.

**Second Session.** In another session the same experiment was repeated with a different thresholds, namely  $\theta = .7$  and  $\epsilon = .01$ , which made the inductive classification much more *cautious*. The outcomes are reported in Table 3.

**Table 2.** Results of the first session with uncertainty threshold  $\theta = .3$  and minimum ball radius  $\epsilon = .1$ : average values  $\pm$  average standard deviations per query

ontology	match rate	commission rate	omission rate	induction rate
SWM	83.99 $\pm$ 01.06	00.00 $\pm$ 00.00	04.80 $\pm$ 00.47	11.21 $\pm$ 00.75
BIOpAX	85.43 $\pm$ 00.43	03.49 $\pm$ 00.23	05.32 $\pm$ 00.02	05.76 $\pm$ 00.25
LUBM	89.77 $\pm$ 00.26	00.00 $\pm$ 00.00	06.68 $\pm$ 00.21	03.55 $\pm$ 00.06
NTN	86.71 $\pm$ 00.32	00.08 $\pm$ 00.00	05.48 $\pm$ 00.21	07.73 $\pm$ 00.33
SWSD	98.12 $\pm$ 00.05	00.00 $\pm$ 00.00	01.30 $\pm$ 00.05	00.58 $\pm$ 00.00
FINANCIAL	90.26 $\pm$ 00.09	04.16 $\pm$ 00.05	02.57 $\pm$ 00.01	03.01 $\pm$ 00.05

**Table 3.** Results of the second session with uncertainty threshold  $\theta = .7$  and minimum ball radius  $\epsilon = .01$ : average values  $\pm$  average standard deviations per query

ontology	match rate	commission rate	omission rate	induction rate
SWM	93.52 $\pm$ 00.58	00.00 $\pm$ 00.00	06.19 $\pm$ 00.59	00.29 $\pm$ 00.05
BIOpAX	81.42 $\pm$ 04.83	00.80 $\pm$ 00.18	13.00 $\pm$ 04.86	04.78 $\pm$ 00.35
LUBM	91.59 $\pm$ 00.24	00.00 $\pm$ 00.00	07.80 $\pm$ 00.23	00.62 $\pm$ 00.02
NTN	83.78 $\pm$ 01.51	00.00 $\pm$ 00.00	14.23 $\pm$ 02.31	01.99 $\pm$ 00.83
SWSD	98.29 $\pm$ 00.05	00.00 $\pm$ 00.00	01.71 $\pm$ 00.05	00.00 $\pm$ 00.00
FINANCIAL	82.65 $\pm$ 00.70	01.56 $\pm$ 00.10	13.72 $\pm$ 00.97	02.08 $\pm$ 00.27

In this session, it is again worthwhile to note that commission errors seldom occurred quite rarely during the various runs, yet the omission error rate is higher than in the previous session (although generally limited and less than 14%). This is due to the procedure becoming more cautious because of the choice of the thresholds: the evidence for a positive or negative classification was not enough to decide on uncertain cases that, with a lower threshold, were solved by the voting procedure. On the other hand, for the same reasons, the induction rate decreased as well as the match rate (with some exceptions). Another difference is given by the increased variance in the results for all indices.

**Third Session.** In yet another session the experiment was repeated with thresholds  $\theta = .5$  and  $\epsilon = .01$ , as a tradeoff between the cautious and the credulous modes. The outcomes are reported in Table 4.

In this case the match and induction rates are better than in the previous sessions. while commission errors are again nearly absent, omission errors decrease w.r.t. the previous session. it is also worthwhile to note that the standard deviations again decrease w.r.t. the outcomes of the previous session.

Summarizing, on a comparison of these tables of outcomes to those reported in previous works on inductive classification with other methods [4, 8], an increase of the performance due to the accuracy of the new method.

Besides, the advantage of this new method is that more parameters are available to be tweaked to get better results depending on the ontology at hand.

**Table 4.** Results of the third session with uncertainty threshold  $\theta = .5$  and minimum ball radius  $\epsilon = .01$ : average values  $\pm$  average standard deviations per query

ontology	match rate	commission rate	omission rate	induction rate
SWM	94.24 $\pm$ 00.83	00.00 $\pm$ 00.00	05.26 $\pm$ 00.86	00.51 $\pm$ 00.24
BioPAX	85.11 $\pm$ 00.95	01.36 $\pm$ 00.29	08.21 $\pm$ 00.90	05.31 $\pm$ 00.44
LUBM	97.49 $\pm$ 00.74	00.00 $\pm$ 00.00	02.47 $\pm$ 00.73	00.04 $\pm$ 00.02
NTN	86.85 $\pm$ 00.24	00.00 $\pm$ 00.00	06.57 $\pm$ 00.74	06.58 $\pm$ 00.63
SWSD	98.29 $\pm$ 00.05	00.00 $\pm$ 00.00	01.71 $\pm$ 00.05	00.00 $\pm$ 00.00
FINANCIAL	87.98 $\pm$ 01.84	03.18 $\pm$ 00.71	06.12 $\pm$ 02.72	02.72 $\pm$ 00.32

Of course, these parameters may be also the subject of a preliminary learning session (e.g. through cross-validation).

Also the elapsed time (not reported here) was comparable, even though a more complex training phase was performed before classification, also similarly to the kernel-based methods [8].

## 5 Concluding Remarks and Possible Extensions

This paper explored the application of a novel similarity-based classification applied to knowledge bases represented in OWL. The similarity measure employed derives from the extended family of semantic pseudo-metrics based on feature committees [4]: weights are based on the amount of information conveyed by each feature, on the grounds of an estimate of its entropy. The measures were integrated in a similarity-based classification procedure that builds models of the search-space based on prototypical individuals. The resulting system was exploited for the task of approximate instance retrieval which can be efficient and effective even in the presence of incomplete (or noisy) information.

An extensive evaluation performed on various ontologies showed empirically the effectiveness of the method and also that, while the performance depends on the number (and distribution) of the available training instances, even working with limited training sets guarantees a good outcomes in terms of accuracy. Besides, the procedure appears also robust to noise since it seldom made critical mistakes.

The utility of alternative methods for individual classification are manifold. On one hand they can be more robust to noise than purely logical methods, and thus they can be exploited in case the knowledge base contains some incoherence which would hinder deriving correct conclusions. On the other hand, instance-based methods are also very efficient compared to the complexity of purely logical approaches. One of the possible applications of inductive methods may regards the task of ontology population [2] known as particularly burdensome for the knowledge engineer and experts. In particular, the presented method may even be exploited for completing ontologies with probabilistic assertions, allowing further more sophisticated approaches to dealing with uncertainty [12].

Extensions of the similarity measures can be foreseen. Since they essentially depend on the choice of the features for the committee, two enhancements may be investigated: 1) constructing a limited number of discriminating feature concepts [6]; 2) investigating different forms of feature weighting, e.g. based on the notion of *variance*. Such objectives can be accomplished by means of machine learning techniques, especially when ontologies with large sets of individuals are available. Namely, part of the entire data can be saved in order to learn optimal feature sets or a good choice for the weight vectors, in advance with respect to their usage.

As mentioned before, largely populated ontologies may also be exploited in a preliminary cross-validation phase in order to provide optimal values for the parameters of the algorithm also depending on the preferred classification mode (cautious or credulous). The method may also be optimized by reducing the set of instances used for the classification limiting them to the prototypical exemplars and exploiting *ad hoc* data structures for maintaining them so to facilitate their retrieval [16].

## References

- [1] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
- [2] Baader, F., Ganter, B., Sertkaya, B., Sattler, U.: Completing description logic knowledge bases using formal concept analysis. In: Veloso, M. (ed.) Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, pp. 230–235 (2007)
- [3] Bloehdorn, S., Sure, Y.: Kernel methods for mining instance data in ontologies. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 58–71. Springer, Heidelberg (2007)
- [4] d'Amato, C., Fanizzi, N., Esposito, F.: Query answering and ontology population: An inductive approach. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 288–302. Springer, Heidelberg (2008)
- [5] Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley, Chichester (2001)
- [6] Fanizzi, N., d'Amato, C., Esposito, F.: Induction of optimal semi-distances for individuals based on feature sets. In: Calvanese, D., et al. (eds.) Working Notes of the 20th International Description Logics Workshop, DL 2007, Bressanone, Italy. CEUR Workshop Proceedings, vol. 250 (2007)
- [7] Fanizzi, N., d'Amato, C., Esposito, F.: DL-FOIL: Concept learning in description logics. In: Železný, F., Lavrač, N. (eds.) ILP 2008. LNCS(LNAI), vol. 5194, pp. 107–121. Springer, Heidelberg (2008)
- [8] Fanizzi, N., d'Amato, C., Esposito, F.: Statistical learning for inductive query answering on OWL ontologies. In: Sheth, A., et al. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 195–212. Springer, Heidelberg (2008)
- [9] Haase, P., van Harmelen, F., Huang, Z., Stuckenschmidt, H., Sure, Y.: A framework for handling inconsistency in changing ontologies. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 353–367. Springer, Heidelberg (2005)

- [10] Hitzler, P., Vrandečić, D.: Resolution-based approximate reasoning for OWL DL. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 383–397. Springer, Heidelberg (2005)
- [11] Huang, Z., van Harmelen, F.: Using semantic distances for reasoning with inconsistent ontologies. In: Sheth, A., et al. (eds.) Proceedings of the 7th International Semantic Web Conference, ISWC 2008. LNCS, vol. 5318, pp. 178–194. Springer, Heidelberg (2008)
- [12] Lukasiewicz, T.: Expressive probabilistic description logics. *Artificial Intelligence* 172(6-7), 852–883 (2008)
- [13] Möller, R., Haarslev, V., Wessel, M.: On the scalability of description logic instance retrieval. In: Parsia, B., Sattler, U., Toman, D. (eds.) Proceedings of the 2006 International Workshop on Description Logics, DL 2006. CEUR Workshop Proceedings, vol. 189. CEUR (2006)
- [14] Tserendorj, T., Rudolph, S., Krötzsch, M., Hitzler, P.: Approximate owl-reasoning with screech. In: Calvanese, D., Lausen, G. (eds.) RR 2008. LNCS, vol. 5341, pp. 165–180. Springer, Heidelberg (2008)
- [15] Wache, H., Groot, P., Stuckenschmidt, H.: Scalable instance retrieval for the semantic web by approximation. In: Dean, M., Guo, Y., Jun, W., Kaschek, R., Krishnaswamy, S., Pan, Z., Sheng, Q.Z. (eds.) WISE 2005 Workshops. LNCS, vol. 3807, pp. 245–254. Springer, Heidelberg (2005)
- [16] Witten, I.H., Frank, E.: *Data Mining*, 2nd edn. Morgan Kaufmann, San Francisco (2005)