Abstract. This work focusses on the problem of clustering resources contained in knowledge bases represented through multi-relational standard languages that are typical for the context of the Semantic Web, and ultimately founded in Description Logics. The proposed solution relies on effective and language-independent dissimilarity measures that are based on a finite number of dimensions corresponding to a committee of discriminating features, that stands for a context, represented by concept descriptions in Description Logics. The proposed clustering algorithm expresses the possible clusterings in tuples of central elements: in this categorical setting, we resort to the notion of medoid, w.r.t. the given metric. These centers are iteratively adjusted following the rationale of fuzzy clustering approach, i.e. one where the membership to each cluster is not deterministic but graded, ranging in the unit interval. This better copes with the inherent uncertainty of the knowledge bases expressed in Description Logics which adopt an open-world semantics. An extensive experimentation with a number of ontologies proves the feasibility of our method and its effectiveness in terms of major clustering validity indices.

Keywords: Clustering, Multi-relational learning, Description Logics, Ontology, Semantic Web Mining

1. Introduction: Clustering in Complex Categorical Spaces

The Semantic Web represents the framework for the next generation of Web applications. Recently, specific multi-relational learning methods are being devised for knowledge bases in the Semantic Web, expressed in the standard representations. Indeed, the most burdensome related maintenance tasks, such
as ontology construction, refinement and evolution, demand such automatization also to enable further Semantic Web applications.

In this work, we focus on unsupervised learning for knowledge bases expressed in such standard languages. In particular, we focus on the problem of clustering semantically annotated resources. The benefits of clustering can be manifold [34]. Clustering annotated resources enables the definition of new emerging concepts (concept formation) on the grounds of the concepts that are already defined in a knowledge base; supervised methods can exploit these clusters to induce new concept definitions or to refining existing ones (ontology evolution); intensionally defined groupings may speed-up the task of approximate search and discovery [5]; a clustering may also suggest criteria for ranking the retrieved resources based on the distance from the cluster centers.

From a technical point of view, most clustering methods are based on the application of similarity (or density) measures defined over a set of attributes of the domain objects. Classes of objects are taken as collections that exhibit low interclass similarity (density) and high intraclass similarity (density). Few methods are able to cope with complex logic-based representations and with respect to some background knowledge that could characterize the resources by using not only their attributes but also the relationships among them. Similarity-based methods have been proposed for both for supervised [8] and unsupervised [21] learning problems in clausal spaces typical of the ILP field [28].

Lately, specific approaches have been proposed that are designed for the typical terminological representations for the Semantic Web, which are ultimately based on the Description Logics [2] (henceforth DLs). The early work on this problem includes logic-based clustering methods were devised for some specific DL languages of limited expressiveness (see [20] and, more recently, [13]). The main drawback of these methods is that they are language-dependent, which prevents them to scale to the standard Semantic Web representations that are mapped on complex DLs. Moreover, purely logical methods can hardly handle noisy data while distance-based ones may be more robust. Hence, from a technical viewpoint, suitable measures for terminological representations mentioned above, which can cope with their semantics, are to be devised. Particularly, a theoretical issue posed comes from the increased indeterminacy yielded by the open-world semantics that is adopted on the knowledge bases, differently from the closed-World semantics which is more generally adopted in other contexts (e.g. database relations and clausal spaces).

These problems lead to the investigation on noise-tolerant and language-independent similarity-based clustering methods for complex representations. Specifically, we resort to an extension of distance-based techniques, which can cope with the standard terminological representations. Indeed, the method is intended for grouping similar resources w.r.t. a notion of similarity, coded in a distance measure, complying with the semantics of knowledge bases expressed in DLs. The individuals are gathered around cluster prototypes (medoids) according to their distance. The choice of the best prototypes is performed through a fuzzy membership approach [3, 7].

The importance of the choice of appropriate metrics for assessing the similarity of resources contained in DL knowledge bases is quite evident. Although some structural dissimilarity measures have been proposed for some specific DLs of fair expressiveness [5], they are still partly based on structural criteria which make them fail to fully grasp the underlying semantics and hardly scale to more complex DL languages such as those backing the OWL ontology language\(^1\). Therefore, we have devised a family of semi-distance measures for semantically annotated resources, which can overcome the aforementioned

\(^1\)http://www.w3.org/TR/owl-guide/
limitations [9, 12]. Such measures are merely based on the criterion of semantic discernibility of the input individuals with respect to a fixed reference context [14] represented by a set of concept definitions. Hence, the new measures are not intended to be absolute, they rather depend on the knowledge base they are applied to. Thus, also the choice of good feature may deserve a preliminary optimization phase, which can be performed by means of stochastic search procedures [12].

In the proposed method, the notion of cluster center, or centroid, characterizing the plethora of distance-based algorithms for numeric representations descending from K-MEANS [18], is replaced by the notion of medoid as a cluster prototype, which fits better categorical representations [19]. Even more so, differently from the deterministic approaches, the proposed clustering algorithm employs a notion of fuzzy membership w.r.t. the current medoids computed according to the measure mentioned above. On each iteration, the choice of medoids evolves by adjusting the cluster-membership probability w.r.t. each medoid.

The type of algorithm had also an impact on the validity indices adopted and modified for the experimental evaluation of the method. Indeed, we present the outcomes of three session of experiments, where the performance of the method is investigated in the problem of clustering resources in real ontologies drawn from common repositories. Each session adopts a different rationale behind the weights assigned to the features, hence we could also investigate the performance w.r.t. the weights tuning.

The remainder of this work is organized as follows. After Sect. 2 that presents the basics of the DL representation, the family of semantic dissimilarity measures adopted by the algorithm are recalled in Sect. 3. The fuzzy clustering algorithm for DL knowledge bases is presented and discussed in Sect. 4. In Sect. 5 we report on the experiments aimed at assessing the validity of the method on a number of ontologies publicly available in the Web. Then, in Sect. 6, we discuss the related work concerning (dis)similarity measures and clustering algorithms on complex logical representations. Conclusions and extensions are finally examined in Sect. 7.

2. Preliminaries on the Representation of the Semantic Knowledge Bases

In the following, we assume that resources, concepts and their relationships are defined in terms of a generic ontology language that may be mapped to some DL language with the standard model-theoretic semantics (see the DLs handbook [2] for a thorough reference). As mentioned in the previous section, one of the advantages of our method is that it does not depend on a specific language for semantic annotations based on DLs. However, the implementation applies to OWL-DL knowledge bases².

In the reference representation, a knowledge base \( \mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A}) \) contains a TBox \( \mathcal{T} \), an RBox \( \mathcal{R} \) and an ABox \( \mathcal{A} \), which are sets of logical axioms using language constructors³ and vocabularies of concept, role and individual names \( \langle \mathcal{N}_C, \mathcal{N}_R, \mathcal{N}_I \rangle \).

\( \mathcal{T} \) is a set of concept definitions: \( C \equiv D \), where \( C \) is the atom denoting the defined concept and \( D \) is a DL concept description specified by the application of the language constructors to primitive concepts and roles. Typical constructors include the conjunction (\( \land \)) and disjunction (\( \lor \)) of simpler concepts. Concept descriptions involving role define new concepts by restricting the domain of a role, such as universal (\( \forall R.C \)), existential (\( \exists R.C \)) and cardinality (\( \leq R.C \), \( \geq R.C \)) restrictions.

²The choice of the languages, that corresponds to the \( SHIQ(D) \) DL was also determined by its decidability [2].
³All these constructors have an equivalent in OWL-DL.
The RBox $\mathcal{R}$ contains similar axioms for specifying new roles by means of proper constructors. The complexity of such definitions depends on the specific DL language. Typical constructors include the conjunction ($\cap$) and disjunction ($\cup$) of simpler roles, inverse roles ($R^-$) and transitivity axioms. Also roles involving concrete domains $\mathcal{D}$ (a.k.a. datatype properties) are allowed in OWL-DL, yet we will not further discuss their usage.

$\mathcal{A}$ contains assertions (ground axioms) on individuals (domain objects) concerning the current world state, namely $C(a)$ (class-membership), $a$ is an instance of concept $C$, and $R(a,b)$ (relations), $a$ is $R$-related to $b$. The set of the individuals referenced in the assertions $\mathcal{A}$ is usually denoted with Ind($\mathcal{A}$). Each individual can be assumed to be identified by a constant (or its own URI in OWL-DL), however this is not bound to be a one-to-one mapping (unique names assumption).

A set-theoretic semantics is generally adopted with these representations, with interpretations $\mathcal{I}$, with their related domain $\Delta^\mathcal{I}$, which map each concept (resp. role) description $C$ (resp. $R$) to its extension, that is a subset of the domain $C^\mathcal{I} \subseteq \Delta^\mathcal{I}$, resp. to $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. Axioms are interpreted as $C^\mathcal{I} \equiv D^\mathcal{I} = (C^\mathcal{I} = D^\mathcal{I})$ (analogously for role axioms). Assertions are simply interpreted as follows: $(C(a))^\mathcal{I} = a^\mathcal{I} \in C^\mathcal{I}$ and $(R(a,b))^\mathcal{I} = (a^\mathcal{I}, b^\mathcal{I}) \in R^\mathcal{I}$. Such interpretations are said to satisfy the axiom and will be called models. The models of interest are those interpretations that satisfy all axioms in the knowledge base $\mathcal{K}$. Entailment will be denoted with $\models$.

In this context the most common inference is the computation of the subsumption relationship between concepts (or roles): given two concept descriptions $C$ and $D$, $D$ subsumes $C$, denoted by $C \subseteq D$, iff for every interpretation $\mathcal{I}$ it holds that $C^\mathcal{I} \subseteq D^\mathcal{I}$. Hence equivalence axioms can be considered as double inclusion axioms. This allows the formation of complex hierarchies of concept/roles.

**Example 2.1. (royal family)**

Suppose $\mathcal{K} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ with

$$\mathcal{T} = \{ \text{Male} \equiv \neg \text{Female}, \text{Woman} \equiv \text{Human} \cap \text{Female}, \text{Man} \equiv \text{Human} \cap \text{Male}, \text{Mother} \equiv \text{Woman} \cap \exists \text{child. Human}, \text{Father} \equiv \text{Man} \cap \exists \text{child. Human}, \text{Parent} \equiv \text{Father} \cup \text{Mother}, \text{Grandmother} \equiv \text{Woman} \cap \exists \text{child. Parent}, \text{MotherWithNoDaughter} \equiv \text{Mother} \cap \forall \text{child. Male} \text{SuperMother} \equiv \text{Mother} \cap \exists \geq 3 \text{child} \}$$

$$\mathcal{R} = \{ \}$$

$$\mathcal{A} = \{ \text{Woman}(\text{ELISABETH}), \text{Woman}(\text{DIANA}), \text{Man}(\text{CHARLES}), \text{Man}(\text{EDWARD}), \text{Man}( \text{ANDREW}), \text{MotherWithNoDaughter}(\text{DIANA}), \text{child}(\text{ELISABETH},\text{CHARLES}), \text{child}(\text{ELISABETH},\text{EDWARD}), \text{child}(\text{ELISABETH},\text{ANDREW}), \text{child}(\text{DIANA},\text{WILLIAM}), \text{child}(\text{CHARLES},\text{WILLIAM}) \}$$

Given this simple knowledge base, it is possible to infer that the inclusions $\text{MotherWithNoDaughter} \sqsubseteq \text{Woman} \sqsubseteq \neg \text{Man}$ hold. Moreover, further facts about the individuals are also derivable through reasoning, such as $\text{Grandmother}(\text{ELISABETH}), \text{Mother}(\text{DIANA})$ and $\text{Father}(\text{CHARLES})$ but not (because of the
Several other inference services are provided by the standard automated reasoners. Like all other instance-based methods, the measures proposed in this section require performing instance-checking, which amounts to determining whether an individual, say \( a \), belongs to a concept extension, i.e. whether \( \mathcal{K} \models C(a) \) holds for a certain concept \( C \). In the simplest cases (primitive concepts) instance-checking requires simple ABox lookups, yet for defined concepts the reasoner may need to perform a number of inferences. It is worthwhile to recall that the Open World Assumption (OWA) is made. Thus, differently from query answering in databases, which boils down to finite model checking, DL reasoning procedures might be unable\(^4\) to ascertain the class-membership or non-membership. Hence one has to cope with this form of uncertainty.

### 3. Metrics for DL: Comparing Individuals within Ontologies

In distance-based cluster analysis, a function for measuring the (dis)similarity of individuals is needed. 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 level before comparing them \([5]\) (resorting to the approximation of the most specific concept of an individual w.r.t. the ABox \([2]\)).

Inspired by some techniques for distance construction and Multi-dimensional Scaling \([7, 16]\), we have proposed the definition of totally semantic distance measures for individuals in the context of a knowledge base which is also able to cope with the OWA. Similar individuals should behave similarly, as concerns their semantics, with respect to similar concepts. We have introduced a novel measure, which is based on the idea of comparing their semantics along a number of dimensions represented by a context \([14]\), i.e. a set of of concept descriptions. Thus, the rationale of the new measure is to compare individuals on the grounds of their behavior w.r.t. a given collection of concept descriptions, say \( \mathcal{F} = \{ F_1, F_2, \ldots, F_m \} \), which stands as a group of discriminating features expressed in the considered DL language.

The general form of the family of dissimilarity measures for individuals inspired by the Minkowski’s distances \( (L_p) \) can be defined as follows \([9, 12]\):

**Definition 3.1. (family of dissimilarity measures)**

Let \( \mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle \) be a knowledge base. Given a set of concept descriptions \( \mathcal{F} = \{ F_1, F_2, \ldots, F_m \} \) and a normalized vector of related weights \( w_i \), a family of dissimilarity measures \( \{ d^F_p \}_{p \in \mathbb{N}} \), with \( d^F_p : \text{Ind}(\mathcal{A}) \times \text{Ind}(\mathcal{A}) \mapsto [0, 1] \), is defined as follows:

\[
\forall a, b \in \text{Ind}(\mathcal{A}) \quad d^F_p(a, b) = \left[ \sum_{i=1}^{m} \left| w_i \cdot (\pi_i(a) - \pi_i(b)) \right|^p \right]^{\frac{1}{p}}
\]

where the projection function vector \( \pi \) is defined \( \forall i \in \{1, \ldots, m\} \)

\(^4\)It may happen that both interpretations satisfying the membership case \( (C(a)) \) and other interpretations satisfying the non-membership case \( (\neg C(a)) \) can be built through tableaux reasoning \([2]\).
Note that the measure is efficiently computed when the feature concepts \( F_i \) are such that the KBMS can directly infer the truth of the assertions \( F_i(a) \), ideally \( \forall a \in \text{Ind}(A) : F_i(a) \in A \). This is very important for the measure integration in algorithms which massively exploit its computation, such as all instance-based methods. The presented method can be regarded as a form of propositionalization [23].

The given definition of the projection functions is basic. The case of \( \pi_i(a) = 1/2 \) corresponds to the case when a reasoner cannot give the truth value for a certain membership query. This is due to the open-world semantics adopted when reasoning with the underlying representation which is suitable for this context.

An intermediate value is just a raw (uniform) estimate of the uncertainty related to the single feature. By properly assigning the weights to vector \( \vec{w} \) it is possible to obtain a measure that better reflects the available knowledge [16]. Even more so, the ugly duckling theorem [7] states that, in absence of specific knowledge, all couples of objects would have the same similarity. Then, in previous works [6, 12], we have proposed setting the weights according to the amount of information conveyed by each feature. This quantity is estimated by the entropy of the features:

\[
H(F_i) = P^1_i \log(1/P^1_i) + P^0_i \log(1/P^0_i) + P^1_i \log(1/P^1_i)
\]

computed from the prior class-membership probability as elicited from the knowledge base:

\[
P^i_a = Pr(\text{check}(a \in F_i) = v)
\]

where the values \( v \) return by the instance-checking reasoning service (check) stand, resp., for non-membership (−1), uncertainty (0) and membership (+1). Then, the weights can be set normalizing the entropies, as follows:

\[
w_i = \frac{H(F_i)}{\sum_j H(F_j)} \quad i = 1, \ldots, m
\]

A further alternative weighting is based on an estimate of the feature variance [12]:

\[
\widehat{\text{var}}(F_i) = \frac{1}{2 \cdot |\text{Ind}(A)|^2} \sum_{a \in \text{Ind}(A)} \sum_{b \in \text{Ind}(A)} [\pi_i(a) - \pi_i(b)]^2
\]

which induces the choice of weights:

\[
w_i = 1/(2 \cdot \widehat{\text{var}}(F_i)) \quad i = 1, \ldots, m
\]

which should be also normalized.

These weights may also be employed to encode also asymmetric cost functions [7, 16] so to assign a different impact to the values that the dissimilarity on a given feature \( ([\pi_i(a) - \pi_i(b)]^p) \) may assume. A more careful choice of the feature concepts to be included in \( \Phi \), based on stochastic search procedure is presented in other works concerning clustering [12].
4. Fuzzy Clustering in Complex Categorical Domains

Many clustering algorithms have been proposed for simple representations (mostly single database tables made up of attribute-value tuples). Most of them are based on distance or density measures and related proximity (density) tables (see [18] for a survey). The schemata of similarity-based clustering algorithms can be adapted to more complex logical settings like the one of interest for this work, especially when similarity or dissimilarity measures for the related instance space are available.

We focus on a generalization of distance-based methods adopting the notion of medoids as cluster centers [19]. The method implements a fuzzy clustering scheme [3, 7], where the representation for the clusterings that are iteratively adjusted is made up of tuples of prototypical individuals for the various clusters but, differently from the \textsc{-means}, the membership of the instances w.r.t. the various clusters is probabilistic rather than deterministic.

4.1. Unsupervised Learning Problem

Clustering of individuals in a knowledge base can be cast as solving an unsupervised learning problem, sketched in Fig. 1.

\begin{itemize}
  \item a knowledge base $\mathcal{K} = \langle T, \mathcal{A} \rangle$ with $\text{Ind}(\mathcal{A})$ is the set of individuals occurring in $\mathcal{A}$
  \item the number of clusters $k$
  \item a global criterion (cost) function $L$
\end{itemize}

\begin{figure*}[h]
\centering
\begin{tabular}{l}
\textbf{given} \\
\hspace{1em} a knowledge base $\mathcal{K} = \langle T, \mathcal{A} \rangle$ with $\text{Ind}(\mathcal{A})$ is the set of individuals occurring in $\mathcal{A}$ \\
\hspace{1em} the number of clusters $k$ \\
\hspace{1em} a global criterion (cost) function $L$
\end{tabular}
\end{figure*}

\begin{figure*}[h]
\centering
\begin{tabular}{l}
\textbf{find} \\
\hspace{1em} a responsibility assignment $r$ of the individuals to a set of $k$ clusters that minimizes $L$
\end{tabular}
\end{figure*}

In case of \textit{hard clustering} one aims at finding $\{C_1, \ldots, C_k\}$ of $k$ subsets of $\text{Ind}(\mathcal{A})$ and a related responsibility function$^5$ $r : \text{Ind}(\mathcal{A}) \times \{1, \ldots, k\} \mapsto \{0, 1\}$, for deciding on the membership of clusters. In case of \textit{soft clustering}, this function can be defined as follows $r : \text{Ind}(\mathcal{A}) \times \{1, \ldots, k\} \mapsto [0, 1]$. Once a soft responsibility assignment has been determined, one may derive a crisp clustering structure by replacing the highest membership value of a pattern with 1 and all others with 0.

This setting may be also made extended by discarding the implicit assumption of disjointness between couples of clusters. However, this setting goes beyond the scope of the present work and will not be further discussed.

4.2. Fuzzy Clustering Algorithm

The algorithm searches the space of possible clusterings of individuals, optimizing a global cost function $L$ based on the relative discernibility of the individuals of the different clusters (inter-cluster separation)

$^5$Sometimes this function is represented as a matrix $[r_{ij}]_{k \times N}$, with $r_{ij} \in [0, 1]$. 


and on the intra-cluster similarity measured in terms of the $d_F$ pseudo-metric. Considered a set of cluster centers (prototypes) $\{\mu_1, \ldots, \mu_k\}$, a notion of graded membership of an individual $x_i$ w.r.t. a given cluster $C_j$ is introduced ranging in $[0, 1]$. This corresponds to computing the responsibility as a probability $r_{ij} = r(x_i, C_j) = \hat{P}(C_j | x_i, \theta)$, where $\theta$ is a vector of parameters for the membership functions.

The objective function to be minimized can be written:

$$L = \sum_{i=1}^{N} \sum_{j=1}^{k} (\hat{P}(C_j | x_i))^{b} \cdot d(x_i, \mu_j)$$

where, for simplicity, we have omitted the dependence on $\theta$ and the free parameter $b$ is used to adjust the blending of the clusters. When $b = 0$, $L$ is merely based on a sum of square errors, with each pattern assigned to a single cluster. If $b > 0$ the criterion function allows each pattern to belong to multiple clusters. The following conditions hold:

$$\sum_{j=1}^{k} \hat{P}(C_j | x_i) = 1 \quad \forall i \in \{1, \ldots, N\}$$
$$0 < \sum_{i=1}^{N} \hat{P}(C_j | x_i) < 1 \quad \forall j \in \{1, \ldots, k\}$$
$$\sum_{i=1}^{N} \sum_{j=1}^{k} \hat{P}(C_j | x_i) = N$$

Its minima are found solving the equations involving the partial derivatives of the criterion function w.r.t. the cluster centers $\mu_j$ and w.r.t. prior the probabilities $\hat{P}_j = P(C_j)$. Then, setting, respectively, $\partial L / \partial \mu_j = 0$ and $\partial L / \partial \hat{P}_j = 0$, we have:

$$\mu_j = \frac{\sum_{i=1}^{N} (\hat{P}(C_j | x_i))^{b} \cdot x_i}{\sum_{i=1}^{N} (\hat{P}(C_j | x_i))^{b}} \quad \forall j \in \{1, \ldots, k\}$$

(1)

and

$$\hat{P}(C_j | x_i) = \left[ \sum_{r=1}^{k} \left( \frac{d_{ij}}{d_{ir}} \right)^{\frac{1}{b-1}} \right]^{-1} \quad \forall i \in \{1, \ldots, N\} \quad \forall j \in \{1, \ldots, k\}$$

(2)

where $d_{ij} = d(x_i, \mu_j)$.

This schema is intended for a numerical setting, such as the DL-FUZZY c-MEANS, where the centers are tuples. For our purposes, we resort to the notion of medoid that was introduced for categorical feature-spaces w.r.t. some distance measure [19, 18]. Namely, the medoid of a group of individuals is an individual that has the minimal average distance w.r.t. the others. Formally:

**Definition 4.1. (medoid)**

Given a set of individuals $S$ and a dissimilarity measure $d$ defined on this set, the medoid of the set is the individual:

$$\mu_S = \text{medoid}(S) = \arg\min_{a \in S} \frac{1}{|S|} \sum_{b \in S} d(a, b)$$

(3)

Thus in our setting, medoids will be chosen as prototypes. However, while the previous definition concerns medoids when the membership of an individual to a certain set is deterministic, in our case graded memberships are to be taken into account. Then, summarizing Eqs. 1 and 3, we can re-define the medoids as follows:
For each cluster $C_j, j \in \{1, \ldots, k\}$:

$$\mu_j = \arg\min_{a \in C_j} \sum_{b \in C_j} d(a, b) \cdot \left( \hat{P}(C_j | a) \right)^b \quad \forall j \in \{1, \ldots, k\}$$

(4)

i.e. the medoids are determined by the individuals minimizing the distance to the other members of the cluster, weighted by their membership probability.

The criterion function is minimized when the cluster centers are close to the individuals that have high (estimated) probability of belonging to the cluster. The classical $k$-MEDOIDS algorithm [19] corresponds to the case when the individuals are assigned to a single cluster: $P(C_j | a) = 1$ if $d(a, \mu_j) < d(a, \mu_l)$ for all $l \in \{1, \ldots, k\}, l \neq j$ and null otherwise. The usage of the graded memberships may improve the convergence of the fuzzy version of the algorithm.

Operationally, a specific dissimilarity measure for individuals like those defined in the previous section can be chosen: $d = d_F^p$ (for some $F$ and $p$). The solutions of the equations above can be estimated iteratively. Fig. 2 reports a sketch of the resulting DL-FUZZY $k$-MEDOIDS algorithm. Note that the algorithm requires the number of clusters $k$ as a parameter.

The representation of centers through medoids has two advantages. First, it presents no limitations on attributes types, and, second, the choice of medoids is dictated by the location of a predominant fraction of elements inside a cluster and, therefore, it is less sensitive to the presence of outliers. This robustness is particularly important in the common context that many elements do not belong exactly to any cluster, which may be the case of the membership in DL knowledge bases, which may be not ascertained given the OWA. Algorithms where prototypes are represented by centroids, which are weighted averages of elements within a cluster, only work conveniently with numerical attributes and can be negatively affected even by a single outlier. An algorithm based on medoids allows for a more flexible definition of similarity. Many clustering algorithms work only after transforming symbolic into numeric attributes.

5. Evaluation

The clustering algorithm has been evaluated with an experimentation on various knowledge bases selected from standard repositories. The option of randomly generating assertions for artificial individuals was discarded for it might have biased the procedure. Only populated ontologies (which may be more difficult to find) were suitable for the experimentation.

5.1. Experimental Setup

A number of different knowledge bases represented in OWL were selected from various sources (the Protégé library\(^6\) and the Swoogle\(^7\) search engine were used), namely: FiniteStateMachines (FSM), SurfaceWaterModel (SWM), Transportation, Wine, NewTestamentNames (NTN), an excerpt of the Financial ontology\(^8\), the glycolysis ontology\(^9\) in the BioPax suite (BioPAX), and one of

\(^6\)http://protege.stanford.edu/plugins/owl/owl-library
\(^7\)http://swoogle.umbc.edu
\(^8\)http://www.cs.put.poznan.pl/alawrynowicz/financial.owl
\(^9\)http://www.biopax.org/Downloads/Level1v1.4/biopax-example-ecocyc-glycolysis.owl
clustering DL-FUZZY-k-MEDOIDS($k$, individuals, maxIterations)

input: $k$: required number of clusters; individuals: individuals to be clustered; maxIterations: maximum number of iterations;
output: clustering: set of clusters

static: iterationNo: iteration counter
{C_j}_{j=1,...,k}: clusters
$\bar{\mu} = (\mu_1, \ldots, \mu_k)$: medoids vector
$\hat{P}(\cdot | \cdot)$: membership probabilities

begin
Initialize iteration $\leftarrow 0$
Initialize random prototypes $\bar{\mu}$
Initialize probabilities $\hat{P}(C_j | x_i)$, for $i = 1, \ldots, N$, $j = 1, \ldots, k$
repeat
For each $a \in$ individuals:
\[ t \leftarrow \arg\min_{j=1,\ldots,k} d(a, \mu_j) \]
\[ C_t \leftarrow C_t \cup \{a\} \]
re-compute cluster prototypes $\bar{\mu}$, using the previous $\hat{P}(C_j | x_i)$ in Eq. (3)
re-compute all probabilities $\hat{P}(C_j | x_i)$ using the new $\bar{\mu}$ in Eq. (1)
for $i = 1, \ldots, N$ do
normalize the probability estimates $\hat{P}(C_j | x_i)$
iterationNo $\leftarrow$ iterationNo + 1
until convergence or iterationNo = maxIterations
return $\{C_j\}_{j=1,...,k}$

end

Figure 2. The fuzzy clustering algorithm for categorical metric spaces.

the ontologies randomly generated by the Lehigh University Benchmark\textsuperscript{10} (LUBM). Table 1 summarizes salient figures concerning these ontologies and their composition.

The most time-consuming preliminary operation was the computation of the proximity matrix based on the adopted measure. For these experiments, the squared version of the measure has been chosen ($d^{L_2}$) with three choices for the weights assignments: uniform, entropic, variance-based (see Sect. 3). All named concepts in the knowledge base have been used for the committee of features, thus guaranteeing meaningful measures with high redundancy. The weights assigned to each feature served to determine their importance for the discrimination of the various individuals. The Pellet reasoner\textsuperscript{11} (ver. 2.0rc7) was employed to perform the inferences that were necessary to compute the proximity matrices.

The indices which were chosen for the experimental evaluation of the outcomes were the following: our fuzzy alternatives for the R-Squared index, Hubert’s normalized $\Gamma$ index and the average Silhou-

\textsuperscript{10}http://swat.cse.lehigh.edu/projects/lubm/
\textsuperscript{11}http://clarkparsia.com/pellet
also an additional (partial) comparison of the outcomes to those obtained with the alternative evolutionary method (not included here as the same datasets yet not the same indices were used).

...through a slower stochastic optimization method based on genetic programming [12]. This would allow weights.

three typologies of measures defined above: uniform weights, entropy-based weights, and variance-based weights.

Finally, the choice of the number of cluster was determined as the average number of clusters found instead of numeric means.

The experimentation consisted of 50 runs of the algorithm per knowledge base. Each run took from less than one second to a few minutes on a QuadCore (2Gb RAM) Linux box, depending on the number of individuals occurring in the processed ontology. Moreover we replicated the experiments adopting the three typologies of measures defined above: uniform weights, entropy-based weights, and variance-based weights.

Finally, the choice of the number of cluster was determined as the average number of clusters found in previous experiments with another clustering algorithm which is able to determine it autonomously through a slower stochastic optimization method based on genetic programming [12]. This would allow also an additional (partial) comparison of the outcomes to those obtained with the alternative evolutionary method (not included here as the same datasets yet not the same indices were used).
5.2. Results

In the following sections we present the outcomes of the experiments with the fuzzy clustering algorithm. In each session, a different choice of the weight vector for the features of the metric is made along with the three different settings presented in Sect. 3, namely uniform, variance-based and entropy-based weighting. Figures in boldface denote the best results observed in the three sessions.

5.2.1. Experiments with the Uniformly Weighted Metric Features

In the first session the weights of the features were intended to be uniformly weighted, then equal relevance was assigned to each of them. Tab. 2 reports the average outcomes per ontology, standard deviations and min-max intervals of indices in the experiments with the mentioned choice for the weight vector of the metric.

Preliminary, from the figures in the table, we observe that the clustering algorithm appears quite stable in terms of all indices, as testified by the low variance of the results, despite its inherent randomized nature. As such, the optimization procedure does not seem to suffer from being caught in local minima.

Table 2. Results of the experiments with the DL-FUZZY k-MEANODS algorithm using the measure adopting uniform weights (± standard deviation and minimum-maximum intervals).

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Hubert’s $\Gamma$</th>
<th>R-Squared</th>
<th>Silhouette</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSM</td>
<td>.46 (±9.11e-2)</td>
<td>.83 (±6.36e-2)</td>
<td>.82 (±2.91e-2)</td>
</tr>
<tr>
<td></td>
<td>[.39,.49]</td>
<td>[.46,.89]</td>
<td>[.79,.91]</td>
</tr>
<tr>
<td>SWM</td>
<td>.76 (±5.21e-2)</td>
<td>.84 (±7.74e-3)</td>
<td>.86 (±5.95e-2)</td>
</tr>
<tr>
<td></td>
<td>[.68,.83]</td>
<td>[.83,.86]</td>
<td>[.81,.95]</td>
</tr>
<tr>
<td>WINE</td>
<td>.32 (±4.30e-2)</td>
<td>.98 (±1.54e-3)</td>
<td>.86 (±1.02e-2)</td>
</tr>
<tr>
<td></td>
<td>[.26,.37]</td>
<td>[.97,.98]</td>
<td>[.83,.88]</td>
</tr>
<tr>
<td>BIO Pax</td>
<td>.55 (±1.16e-1)</td>
<td>.53 (±1.07e-1)</td>
<td>.88 (±1.51e-2)</td>
</tr>
<tr>
<td></td>
<td>[.43,.76]</td>
<td>[.45,.69]</td>
<td>[.85,.92]</td>
</tr>
<tr>
<td>NTN</td>
<td>.76 (±5.22e-2)</td>
<td>.84 (±7.74e-3)</td>
<td>.86 (±5.95e-2)</td>
</tr>
<tr>
<td></td>
<td>[.68,.83]</td>
<td>[.83,.86]</td>
<td>[.82,.95]</td>
</tr>
<tr>
<td>FINANCIAL</td>
<td>.45 (±1.81e-2)</td>
<td>.50 (±1.30e-2)</td>
<td>.86 (±1.20e-2)</td>
</tr>
<tr>
<td></td>
<td>[.41,.46]</td>
<td>[.43,.50]</td>
<td>[.80,.89]</td>
</tr>
<tr>
<td>LUBM</td>
<td>.37 (±7.73e-2)</td>
<td>.13 (±3.84e-2)</td>
<td>.88 (±3.93e-2)</td>
</tr>
<tr>
<td></td>
<td>[.14,.47]</td>
<td>[.02,.20]</td>
<td>[.83,.95]</td>
</tr>
</tbody>
</table>

Considering the normalized Hubert’s $\Gamma$ index, which measures both compactness and separation of the resulting clusters w.r.t. the proximity matrix, the results are generally fairly good for all ontologies, also in respect of the range of this measure, with some cases (SWM and NTN) where they reach .76 on average. The lowest figures for WINE (.32) and LUBLM (.37) may be explained by the large number of different concepts and the number of individuals, respectively.
The R-Squared average outcomes denote a good degree of separation between the various clusters, but for the case of LUBM because of many runs with unfortunate choices of initial medoids. We may interpret the outcomes observing that clusters generally show a high degree of compactness. It should also be pointed out that flat clustering penalizes separation as the concepts in DL knowledge bases have to be explicitly declared to be disjoint. Rather, they naturally tend to form subsumption hierarchies.

As concerns the average Silhouette index the performance of the algorithm is generally very good for all ontologies taken into account.

5.2.2. Experiments with Variance-Weighted Metric Features

In the second session the weights of the features were weighted according to an estimate of their variance, then a different relevance was reflecting the ability to discriminate different individuals. Tab. 3 reports the average outcomes of the experiments where variance-based weighted features have been adopted.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Hubert’s $\Gamma$</th>
<th>R-Squared</th>
<th>Silhouette</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSM</td>
<td>.45 (±3.11e-2) [.40,.52]</td>
<td>.76 (±3.56e-3) [.76,.77]</td>
<td>.88 (±6.19e-3) [.88,.90]</td>
</tr>
<tr>
<td>SWM</td>
<td>.78 (±1.56e-2) [.76,.83]</td>
<td>.96 (±1.27e-3) [.96,.97]</td>
<td>.83 (±5.18e-3) [.83,.84]</td>
</tr>
<tr>
<td>WINE</td>
<td>.40 (±3.77e-3) [.40,.43]</td>
<td>.98 (±7.21e-4) [.978,.983]</td>
<td>.92 (±1.81e-3) [.91,.92]</td>
</tr>
<tr>
<td>BIOPAX</td>
<td>.69 (±4.89e-3) [.68,.70]</td>
<td>.74 (±3.49e-16) [.7435,.7435]</td>
<td>.85 (±7.97e-4) [.85,.86]</td>
</tr>
<tr>
<td>NTN</td>
<td>.76 (±2.03e-2) [.68,.78]</td>
<td>.86 (±1.67e-2) [.82,.91]</td>
<td>.92 (±1.12e-2) [.89,.94]</td>
</tr>
<tr>
<td>FINANCIAL</td>
<td>.65 (±2.40e-3) [.65,.66]</td>
<td>.57 (±5.61e-16) [.5728,.5728]</td>
<td>.91 (±1.78e-4) [.905,.906]</td>
</tr>
<tr>
<td>LUBM</td>
<td>.64 (±5.49e-4) [.65,.65]</td>
<td>.59 (±6.43e-4) [.58,.59]</td>
<td>.90 (±9.26e-4) [.89,.90]</td>
</tr>
</tbody>
</table>

Overall, also in this case the algorithm proves even more stable (see the standard deviation of the R-Squared index for the BIOPAX and FINANCIAL ontologies). Moreover, there is a general improvement in terms of all indices, especially in the cases that were found to be more difficult when adopting the uniformly weighted metric. This leads to conclude that a fine-tuning of the weights is essential for the employment of the measure.

More specifically, the average $\Gamma$ index measures were improved in almost all cases and especially with the ontologies containing more individuals, which probably allowed for better estimates of the feature variance for the metric weights. The same considerations apply also for the R-Squared index.
average values which gained an +.46 in the case of the most populated ontology (LUBM). The Silhouette index average values remain quite high and in some cases get to .90 and above.

5.2.3. Experiments with Entropy-Weighted Metric Features

In the third session the weights of the features were weighted according to an estimate of their entropy, then an information measure was used in this case to reflect the ability to discriminate different individuals. Tab. 4 reports the average outcomes of the experiments where entropy-based weighted features have been adopted.

Table 4. Results of the experiments with the DL-FUZZY \textit{k}-MEDOIDS algorithm using the measure adopting weights based on the concept entropy (± standard deviation and minimum-maximum intervals).

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Hubert's $\Gamma$</th>
<th>R-Squared</th>
<th>Silhouette</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSM</td>
<td>.68 (±.921e-3)</td>
<td>.99 (±.374e-3)</td>
<td>.99 (±.595e-3)</td>
</tr>
<tr>
<td></td>
<td>[.67,.70]</td>
<td>[.99,.10]</td>
<td>[.98,.10]</td>
</tr>
<tr>
<td>SWM</td>
<td>.78 (±.196e-2)</td>
<td>.95 (±.221e-3)</td>
<td>.84 (±.495e-3)</td>
</tr>
<tr>
<td></td>
<td>[.75,.84]</td>
<td>[.94,.95]</td>
<td>[.82,.84]</td>
</tr>
<tr>
<td>WINE</td>
<td>.40 (±.377e-3)</td>
<td>.98 (±.733e-4)</td>
<td>.91 (±.188e-3)</td>
</tr>
<tr>
<td></td>
<td>[.40,.43]</td>
<td>[.97,.98]</td>
<td>[.90,.92]</td>
</tr>
<tr>
<td>BIOPAX</td>
<td>.69 (±.594e-3)</td>
<td>.72 (±.139e-2)</td>
<td>.85 (±.811e-4)</td>
</tr>
<tr>
<td></td>
<td>[.68,.71]</td>
<td>[.7238,.7239]</td>
<td>[.851,.855]</td>
</tr>
<tr>
<td>NTN</td>
<td>.80 (±.216e-2)</td>
<td>.83 (±.124e-2)</td>
<td>.94 (±.871e-3)</td>
</tr>
<tr>
<td></td>
<td>[.74,.81]</td>
<td>[.81,.86]</td>
<td>[.91,.94]</td>
</tr>
<tr>
<td>FINANCIAL</td>
<td>.66 (±.277e-3)</td>
<td>.56 (±.494e-16)</td>
<td>.91 (±.176e-4)</td>
</tr>
<tr>
<td></td>
<td>[.66,.67]</td>
<td>[.5640,.5640]</td>
<td>[.906,.907]</td>
</tr>
<tr>
<td>LUBM</td>
<td>.64 (±.755e-4)</td>
<td>.64 (±.619e-4)</td>
<td>.89 (±.926e-5)</td>
</tr>
<tr>
<td></td>
<td>[.641,.645]</td>
<td>[.63,.64]</td>
<td>[.8934,.8935]</td>
</tr>
</tbody>
</table>

Overall, also in this case the algorithm proves to be very stable and another slight general improvement in terms of all indices, especially in the cases that were found to be more difficult when adopting the uniformly weighted metric.

More specifically, the performance confirmed the general improvement, w.r.t. the first session, in terms of the average $\Gamma$ index adding some further positive variation. The same considerations apply also for the R-Squared especially for the smaller ontologies (in terms of their population). The Silhouette index average values are again quite high and in some cases go far beyond the .92 which was the best result observed in the previous session. We may conclude that the variance-based weighting tends to increase the performance in terms of R-Squared index, the entropy-based weighting yielded better outcomes in terms of Hubert’s index, and they have almost the same effect in terms of Silhouette.

Again a fine-tuning of the weights yielded an improvement of the metric effectiveness. The strategy pursued here of finding a good weighting of the existing features, instead of a more complex feature
construction algorithm investigated in previous works [12], may lead to results that are often comparable yet they can be obtained with a less computationally expensive preliminary session.

6. Related Work

The unsupervised learning procedure presented in this paper is mainly based on two factors: the semantic dissimilarity measure and the clustering method. To the best of our knowledge in the literature there are very few examples of similar clustering algorithms working on complex representations that are suitable for knowledge bases in the Semantic Web context. Thus, in this section, we briefly discuss sources of inspiration for our procedure and some related approaches.

6.1. Similarity Measures for Logical Representations

As previously mentioned, various attempts to define semantic similarity (or dissimilarity) measures for concept languages or other fragments of FOL that are similar to DLs, have been made. However applicability of the proposed measures is still limited to simple languages [4] or they are not completely semantic depending also on the structure of the descriptions, on normal forms and on operations that can map individuals to very specific concepts [5]. Very few works investigate on measures that are able to compare individuals directly rather than concepts. A very recent work investigated the problem of individual similarity in the context of recommender systems [1].

In the context of clausal logics, a metric was defined [29] for the Herbrand interpretations of logic clauses as induced from a distance defined on the space of ground atoms. This kind of measures may be employed to assess similarity in deductive databases. Although it represents a form of fully semantic measure, the underlying semantics is different from the standard ones adopted for DL knowledge bases. Therefore the transposition of the proposed measure to the context of interest is not straightforward.

Our measures are mainly based on Minkowski’s norms [36] and on a method for distance induction developed in the context of machine learning [16]. A source of inspiration was also rough sets theory [31] which aims at the formal definition of vague sets by means of their approximations determined by an indiscernibility relationship. Hopefully, methods developed in this context may help solving further open points in our framework and suggest new ways to deal with uncertainty.

6.2. Clustering Procedures

The clustering algorithm adapts the distance-based approach (see [18]) to the logical representations devised for the SW context. Specifically, many methods derived from K-MEANS and K-MEDOIDS (and PAM) represent each cluster with one prototypical element and the other elements are assigned according to a minimal distance criterion (that minimizes a global cost function).

CLARA [19] and CLARANS [27] represent other early exemplars of this approach which implement iterative optimization methods that relocate elements between perspective clusters and recompute medoids. A graph is considered whose nodes are sets of k medoids and an edge connects two nodes if they differ by one medoid. While CLARA compares very few neighbors (a fixed small sample), CLARANS, starting with an arbitrary node, generates neighbors through a stochastic search and checking a maximal number of neighbors. If a neighbor represents a better clustering, the process continues
with this new node. Otherwise a local minimum is found, and the algorithm restarts until a certain number of local minima is found. The best node is returned for the formation of the final partition.

Further related clustering approaches are those based on an indiscernibility relationship [17]. While in our method this idea is embedded in the semi-distance measure (and the choice of the committee of concepts), these algorithms are based on an iterative refinement of an equivalence relationship which eventually induces clusters as equivalence classes.

Our algorithm may be considered a fuzzy extension of the traditional crisp clustering methods where the semantic distance measure allows for the adaptation to logical settings. Exploiting a similar metric, we have proposed [11] a hierarchical extension of PAM working on resources contained in DATALOG knowledge bases. A similar approach was followed in [21], where a distance measure devised for clausal spaces [8] was exploited for clustering.

As mentioned in the introduction, the classic approaches to conceptual clustering [34] in complex (multi-relational) spaces are based on both structure and semantics of the concept and individual descriptions. Kietz & Morik [20] proposed a method for efficient construction of knowledge bases for the BACK terminological language [2]. This method exploits the assertions concerning the roles available in the knowledge base to assess, in the corresponding relationships, subgroups of the domains and ranges which may be inductively deemed as disjoint. In the successive phase, supervised learning methods are used on the discovered disjoint subgroups to construct new concepts that account for them. A similar approach is adopted in [13], where the unsupervised phase is followed a supervised one, performed as an iterative refinement step, exploiting suitable refinement operators for a different DL, namely \( \mathcal{ALC} \) [2]. More recently we developed also an extension of the partitional clustering techniques, such as the \textsc{Bisection Around Medoids} algorithm for DL representations, which exploits similar metrics [9]. This algorithm descends from the (\textsc{Bisection}) \textsc{K-Means} [18] originally developed for numeric or ordinal features. Namely, the worst cluster (in terms of some criterion-index) is divided into two sub-clusters using a \textsc{K-Medoids} procedure \((k = 2)\). The algorithm ends when the required number of clusters is obtained.

7. Concluding Remarks and Extensions

This work has presented a framework for fuzzy clustering that can be applied to standard multi-relational representations adopted for knowledge bases in the Semantic Web context. Its intended usage is for discovering interesting groupings of semantically annotated resources. The clustering algorithm is an extension of standard distance-based clustering procedures employing medoids as cluster prototypes so to deal with complex representations of the target context. Fuzzy clustering may be more suitable to capture the inherent uncertainty underlying representations which adopt an open-world semantics.

The method can be applied to a wide range of concept languages as it exploits a family of metrics that is based on the underlying semantics w.r.t. a context, corresponding to a committee of features represented by a group of concept descriptions in the language of choice. A preliminary learning phase may optimize the choice of the most discriminating features by assigning different weights reflecting their discriminative power. Extensive experiments with various choices of weights demonstrated that this may be almost as effective as adopting a more complex (and computationally demanding) feature construction method as done in previous works [12].
Validity indices for crisp clusterings have been adapted to a fuzzy-clustering setting in order to measure the quality of the resulting partitions. However we probably need to resort to modifications of other indices thought for fuzzy-clustering on numeric spaces [35].

**Conceptual Clustering.** Conceptual Clustering [34] concerns the task of building an intensional justification accounting for the elements that belong to a cluster (and do not belong to others). Hence it adopts supervised learning techniques that should be able to induce new concept descriptions arising from the clustering structures. This was tackled in the context of terminological representations in previous papers [20, 13]. Newly proposed supervised learning methods [24, 10] may suggest better conceptual clustering approaches. Even more so, coupling probabilistic membership functions may be exploited for statistical relational learning oriented to probabilistic extensions of the ontology representations [22, 25].

**Ontology Evolution.** Evolutionary clustering methods are often able determine an optimal number of clusters autonomously. Alternatively, this number is determined by an ad-hoc stochastic procedures such as the Chinese Restaurant Process [32]. Following our investigation of genetic programming [12] other methods may considered such as those based on niching [26]. A natural extension of the clustering algorithms is towards various forms of incrementality. In the case of ontologies, this may be intended as a process of collocation of newly available resources in the correct place within the concept hierarchy but this may also give rise to refinements of the existing concepts or to new concepts / relationships (see paragraph above). The most important problems related to ontology evolution [30] are the concept drift and novelty detection. The method implemented in the system OLINDDA [33] for simple representations has suggested a way for using clustering for tackling these problems in a ontological context [9].

**Hierarchical Clustering.** Another natural extension may concern upgrading the algorithm so that it may build hierarchical clusterings level-wise in order to produce (or reproduce) terminologies, possibly introducing new concepts elicited from the ontology population. Hierarchical clustering methods may adopt agglomerative (clumping) or divisive (splitting) approaches and usually require distance functions for calculating distance between clusters. Given a partitional flat clustering algorithm like the one presented in this paper, hierarchical clustering methods are quite easy to define. One begins with one large cluster and then iteratively breaks these clusters into smaller and smaller ones until a stopping criterion is met (no quality improvement or singleton clusters reached). The clusters at each level are examined and the one containing objects that are the farthestmost according to the given metric are broken apart. Finally a dendrogram is produced. On each level, the worst cluster is selected on the grounds of its quality, e.g. the one endowed with the least average inner similarity (or cohesion). This cluster is candidate to being split. The partition is constructed by calling a partitional clustering method like DL-FUZZY K-MEDOIDS on the worst cluster. In the end, the candidate cluster is replaced by the newly found parts at the next level of the dendrogram.

**Semantic Retrieval.** Another promising research line regards the usage of the metric and the outcomes of clustering for enabling retrieval by analogy procedures [5]: a search query may be issued by means of prototypical resources; answers may be retrieved based on local models (intensional concept descriptions) for the prototype constructed (on the fly) based on the most similar resources (w.r.t. the metric).
Approximate classification of individuals w.r.t. to query concepts can be exploited also for matchmaking. The distance measure may also serve as a ranking criterion.

References


