On the Generalization of the Discovery of Subsumption Relationships to the Fuzzy Case

Fernando Bobillo, Carlos Bobed and Eduardo Mena
Aragon Institute of Engineering Research (I3A), University of Zaragoza, Spain
Email: {fbobillo, cbobed, emena}@unizar.es

***Abstract*—Many real-world applications require ontology alignment to integrate semantic information from different sources. However, most of the work in the field is restricted to finding synonymy relationships, and hyponymy relationships have not received similar attention. In this paper, we discuss some extensions of our previous work in the discovery of subsumption relationships with fuzzy clustering, aggregation operators, Formal Concept Analysis, and synonymy relationships.**

I. Introduction

In recent years, *ontologies* have become a standard for knowledge representation. An ontology is an explicit and formal specification of the concepts, individuals, and relationships that exist in some area of interest, created by defining axioms that describe the properties of these entities [1]. They have been successfully used as part of expert and multiagent systems, as well as a core element in the Semantic Web. Ontologies allow adding semantics to data, making knowledge maintenance, information integration, and reuse of components easier. However, there are still unsolved interoperability problems when dealing with terms taken from different ontologies as we do not usually know how they relate to each other.

In this context, *ontology alignment* consists in finding semantic relationships between elements belonging to different ontologies [2]. For example, it is common to look for synonymy, hyponymy, or disjointness relations between a concept from a source ontology and a concept from a target ontology. Using the resulting alignments, the integration of different ontologies becomes easier, partially solving the above mentioned interoperability problems. Ontology alignment is widely recognized as a very important problem for data integration from different sources, and we find it particularly interesting in semantic mobile distributed systems. For example, *semantic apps* using semantic reasoners on mobile devices [3] typically need to integrate the user context (usually represented using an ontology) with more general domain ontologies or, in multiagent scenarios, with ontological knowledge from other users that co-operate with him to solve complex tasks.

Although there has been a considerable amount of work in the field of ontology alignment, most of the approaches restrict restrict themselves to the problem of finding synonymy relationships (i.e., finding pairs of concepts from different ontologies such that are semantically equivalent). We are however more interested in the less studied problem of finding hyponymy relationships (i.e., finding pairs of concepts from different ontologies such that one of them is a strict subclass of the other one). For example, the Ontology Alignment Evaluation Initiative (OAEI)[1] organizes on an annual basis a benchmark of ontology alignment systems, mainly focusing on synonymy relationships, but only two of these editions (2009[2] and 2011[3]) included an "oriented matching" track dedicated to hyponymy relationships. Synonymy is a very strict relationship that implies that the two aligned entities have exactly the same meaning: two equivalent concepts must have exactly the same instances. On the contrary, in real world domains it is more common to find terms that are quite similar but not exactly the same, as it happens with hyponymy, where two related concepts share some instances but not all of them.

In the Distributed Information Systems research group of the University of Zaragoza[4], we have some previous work in the area. More precisely, some members of our group presented an approach for the extraction of subsumption relationships among concepts from different ontologies [4]. The objective of this paper is to continue this research line by presenting some extensions of our previous work. In particular, we will discuss how the use of fuzzy logic (more precisely, fuzzy clustering and aggregation operators) brings many benefits at different steps of the approach. Furthermore, we will discuss the use of Formal Concept Analysis and synonymy relationships, as well as some repeatability issues.

The rest of this paper is organized as follows. Firstly, Section II recalls how the previous approach in [4] works. Then, we discuss two different generalizations to the fuzzy case: by using fuzzy clustering (Section III) and by using fuzzy aggregation operators (Section IV). Section V describes other extensions and contributions not related to fuzzy logic. Finally, Section VI overviews some related work and Section VII sets out some conclusions and ideas for future work.

II. Background

This section quickly overviews the previous work in [4]; the interested reader is referred there for details and formulae. That work computes subsumption relationships of the form $s \sqsubseteq S$ where one of them is a concept from a source ontology and the other one is a concept from a target ontology. The approach includes the following steps to discover subsumption relationships (as illustrated in Figure 1):

[1]http://oaei.ontologymatching.org
[2]http://oaei.ontologymatching.org/2009/oriented
[3]http://oaei.ontologymatching.org/2011/oriented
[4]http://sid.cps.unizar.es

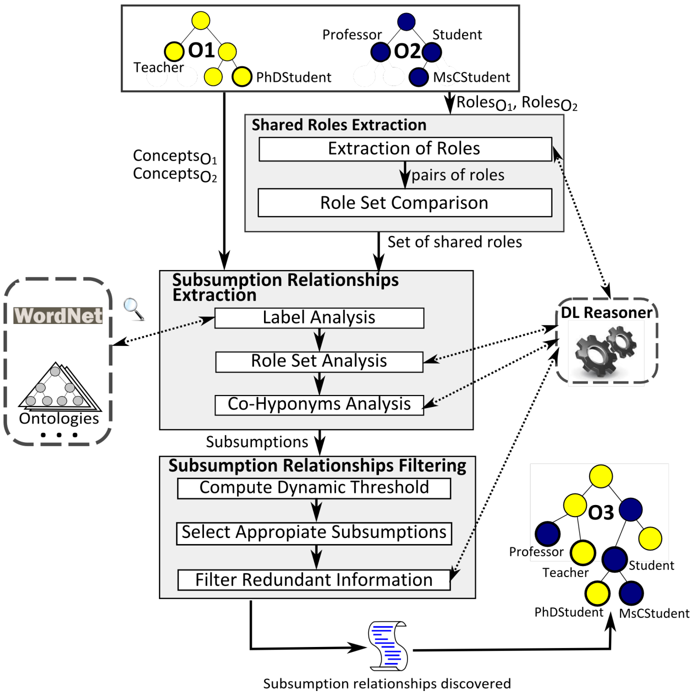

Fig. 1. Architecture to extract subsumption relationships from two ontologies.

- *Extraction of the set of shared roles* R_C for each concept C (those having C as implicit/explicit domain/range). An ontology reasoner is used to discover implicit knowledge.
- *Subsumption relationships extraction* using the concept ontological context (labels, role set analysis, potential co-hyponym, ...) to compute the subsumption degrees.
- *Subsumption relationships filtering* using a dynamic threshold to discard the less probable relationships.

The second step is the most complicated one. The subsumption degree between two concepts s, S from a source and a target ontology is computed as:

$$w_l \cdot sd(l_s, l_S) + w_r \cdot sd(R_s, R_S) + w_{ch} \cdot sd(R_s, hypo_S) \quad (1)$$

where $sd(l_s, l_S)$, $sd(R_s, R_S)$, and $sd(R_s, hypo_S)$ denote the subsumption degrees between their labels, roles, and co-hyponyms (respectively), and $w_l, w_r, w_{ch} \in [0, 1]$ are weights such that $w_l + w_r + w_{ch} = 1$. One of the problems of the approach is the high number of parameters to be tuned (sd functions also depend on some inner parameters). The subsumption degree between concept labels is computed combining information about their relationships from third-party lexical databases and their similarity string metric. The subsumption degree between roles is based on intuitive ideas such as the *duck test* (*"if it looks like a duck, swims like a duck, and quacks like a duck, then it probably is a duck"*), which implies for example that the subsumption degree is proportional to the percentage of roles of S that s has, and to the number of shared roles, or the *opposite duck test* (*"if it does not look like a duck, does not swim like a duck, and does not quack like a duck, then it probably is not a duck"*), which implies for example that if no roles are shared, the subsumption degree is inversely proportional to the number of roles of S. Finally, the subsumption degree between co-hyponyms is based on the idea that if a concept C shares roles with a concept s which is known to be a hyponym of S, then it is more likely to be a hyponym of S as well.

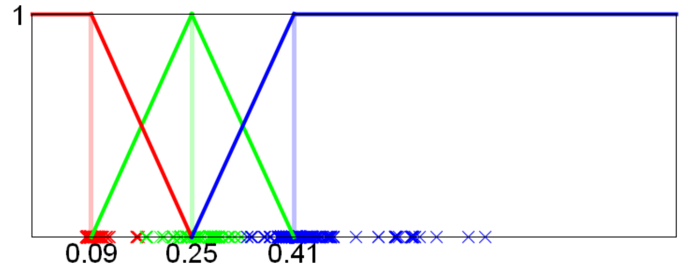

Fig. 2. Clusters obtained in the case 101–222: each subsumption degree is colored as very probable (blue), doubtful (green) or clearly unrelated (red).

III. Extension with Fuzzy Clustering

One of the most interesting features of the previous approach to discover subsumption relationships in [4] is the possibility of select *dynamic thresholds*, that is, given a set of computed alignments between two ontologies, the approach is able to automatically choose a threshold so that only the best alignments (those with a degree greater or equal than the threshold) are actually kept. Indeed, practice shows that it is important to associate different thresholds depending to the different input ontologies: a suitable threshold for a pair of ontologies might be too high or too low for another couple.

It is also evident that for practical reasons to automate this process as far as possible is very important. On the one hand, it is difficult for a human to decide which is the best threshold; on the other hand, even for an expert, this is not a feasible option if a lot of thresholds need to be computed, as it happens during the evaluation of an alignment software, or in ontology mapping competitions.

In [4] the authors group all the data (subsumption degrees between pairs of concepts) into 3 clusters (very probable, doubtful, and clearly unrelated degrees), using *k-means*. Then, the minimum datum of the cluster with a higher centroid is chosen as a threshold to filter the results. The implementation uses the Java Machine Learning Library (Java-ML)[5].

Rather than using a crisp clustering algorithm, we propose to use a *fuzzy c-means algorithm* [5]. The fuzzy c-means algorithm groups a set of n data x_j into c clusters described by means of their centroids c_i (see Figure 2 for an example). Then, the higher centroid can be selected as a threshold.

The main difference in fuzzy c-means algorithm is that every datum can belong to several clusters with different degrees of membership. To this end, the algorithm considers a matrix of membership degrees: μ_{ij} denotes the membership degree of the datum x_j to the i-th cluster. More precisely, the steps of the algorithm are the following ones:

[5]http://java-ml.sourceforge.net

1) Initialize the matrix of membership degrees by assigning to each μ_{ij} a random number in $[0,1]$.
2) Compute the c centroids: $c_i = (\sum_{j=1}^{n} \mu_{ij}^{m} x_j) / \sum_{j=1}^{n} \mu_{ij}^{m}$
3) Update the matrix of membership degrees as follows:

$$\mu_{ij} = \Big(\sum_{k=1}^{c} \frac{\|x_j - c_i\|^{2/(m-1)}}{\|x_j - c_k\|^{2/(m-1)}} \Big)^{-1}$$

4) If the stop condition does not hold, go to step 2.

Initially, we tried to use an existing implementation, the library *Jminhep*[6]. However, we obtained very often unexpected results such as negative values. Since there is not enough support or documentation, we used our own implementation of the algorithm with the following parameters:

- Fuzziness: $m = 2$.
- Stop condition: 1 iteration without changes in centroids.
- Precision for the centroid similarity: 0.0001.

Using fuzzy c-means in our case brings several advantages.

Firstly, it is well-known that fuzzy c-means is usually more stable to the random choice of the initial centroids, and we have verified that this also happens in our case. Indeed, the implementation used in [4] computes 15 repetitions of the k-means algorithm and returns the maximum of the thresholds obtained in each repetition. By using fuzzy c-means we can significantly reduce the number of repetitions: the differences are so small that a single execution is enough. Table I shows the results of some experiments involving the OAEI 2009 oriented track benchmark, providing reference alignments (or official results) between a source ontology 101 and several target ones. In particular, we show maximal value, minimal value, maximal difference, and standard deviation for the thresholds obtained in the fuzzy and the crisp case. In fact, the average maximal difference between different executions of the crisp clustering algorithm is 0.14 (but can reach 0.257 in some cases, for ontologies 101 and 201), whereas in the fuzzy algorithm it is 0.005. To illustrate the importance of such differences, let us consider the case of 101 and 302, where the maximal difference is similar to the average value (0.125). The minimal threshold produces 58 candidates to be subsumption degrees, later filtered to 12, whereas the maximal threshold produces 288, later filtered to 20.

Secondly, since fuzzy clustering does not require to repeat multiple executions, it reduces the global running time.

Thirdly, [4] assumes that there will always be some good alignments. If the two compared ontologies are completely unrelated, so there should be no alignments at all, the approach would return the "least bad" thresholds. By using fuzzy c-means, we know the membership degrees of each datum to all the clusters, so we can detect these situations: if the distance between two centroids is very small, a degree can be classified as a very probable relationship or as a doubtful one, so we can assume that this is one of those cases where there are no good thresholds. For example, in the completely unrelated ontologies 101 and 102 the smallest difference between two centroids is only 0.006 (compare it with the alignment of 101 and 222, where the smallest difference is 0.16).

[6]http://jwork.org/jminhep

TABLE I
THRESHOLDS OBTAINED USING CLASSICAL AND FUZZY C-MEANS.

	Fuzzy clustering				Classic clustering			
Ontologies	Max	Min	Dif	σ	Max	Min	Dif	σ
101–102	0.113	0.113	0	0	0.130	0.113	0.017	0.008
101–103	0.330	0.330	0	0	0.501	0.330	0.171	0.083
101–104	0.325	0.325	0	0	0.492	0.325	0.167	0.043
101–201	0.334	0.334	0	0	0.591	0.334	0.257	0.118
101–202	0.149	0.149	0	0	0.215	0.130	0.085	0.039
101–203	0.335	0.335	0	0	0.511	0.335	0.176	0.062
101–204	0.334	0.334	0	0	0.508	0.334	0.174	0.080
101–205	0.334	0.334	0	0	0.495	0.334	0.161	0.067
101–206	0.326	0.326	0	0	0.496	0.334	0.161	0.057
101–207	0.326	0.326	0	0	0.496	0.334	0.161	0.082
101–208	0.304	0.304	0	0	0.455	0.306	0.149	0.073
101–209	0.201	0.201	0	0	0.201	0.130	0.071	0.029
101–210	0.197	0.197	0	0	0.267	0.197	0.069	0.024
101–222	0.334	0.334	0	0	0.503	0.334	0.169	0.086
101–223	0.326	0.326	0	0	0.478	0.329	0.150	0.068
101–224	0.334	0.334	0	0	0.501	0.334	0.167	0.069
101–225	0.334	0.334	0	0	0.501	0.334	0.167	0.063
101–230	0.309	0.308	0	0	0.424	0.270	0.154	0.071
101–231	0.334	0.334	0	0	0.501	0.334	0.167	0.085
101–237	0.334	0.334	0	0	0.507	0.507	0	0
101–238	0.328	0.328	0	0	0.215	0.328	0.150	0.038
101–249	0.149	0.149	0	0	0.206	0.149	0.066	0.017
101–251	0.149	0.149	0	0	0.206	0.129	0.076	0.029
101–252	0.149	0.149	0	0	0.206	0.149	0.057	0.024
101–258	0.149	0.149	0	0	0.206	0.129	0.076	0.027
101–259	0.146	0.146	0	0	0.206	0.168	0.038	0.010
101–301	0.330	0.220	0.110	0.045	0.330	0.186	0.144	0.070
101–302	0.176	0.176	0	0	0.240	0.115	0.125	0.049
101–303	0.107	0.065	0.043	0.019	0.113	0.113	0	0
101–304	0.138	0.138	0	0	0.218	0.148	0.070	0.018

Fourthly, fuzzy c-means can produce a more accurate threshold in terms of precision, recall, and F-measure. In our case, these values were computed as follows (where $retrievedAlignments$ and $relevantAlignments$ denote sets of hyponymy relationships retrieved by our system or in the OAEI 2009 oriented track reference alignments, respectively):

$$\text{precision} = \frac{\mid \text{relevantAlignments} \cap \text{retrievedAlignments} \mid}{\mid \text{retrievedAlignments} \mid}$$

$$\text{recall} = \frac{\mid \text{relevantAlignments} \cap \text{retrievedAlignments} \mid}{\mid \text{relevantAlignments} \mid}$$

$$\text{F} = (2 \cdot \text{precision} \cdot \text{recall}) / (\text{precision} + \text{recall})$$

Table II shows the detailed results, where the color green indicates that fuzzy clustering outperforms crisp clustering significantly (a bigger difference than 0.5), red means that crisp clustering outperforms fuzzy clustering, and black denotes a tie. Recall increased in 24 % of the cases; in the rest, there are no important differences.

Unfortunately, precision and F-measure can actually decrease, although this did not happen in 70 % and 73 % of the cases, respectively. We think that the reason of this decrease can be that the subsumption degrees that are currently computed are not precise enough; future work will include a deeper study of this behavior. As a final note, fuzzy clustering can produce either higher or lower thresholds, and thus more or less candidates to be subsumption relationships, but in our experiments it always produced more alignments.

TABLE II
PRECISION, RECALL, AND F-MEASURE OF BOTH CLUSTERINGS.

	Fuzzy clustering			Classic clustering		
Ontologies	Precision	Recall	F-measure	Precision	Recall	F-measure
101-102	1	1	1	0	1	0
101-103	0.28	0.2	0.23	0.28	0.1	0.15
101-104	0.28	0.2	0.23	0.28	0.1	0.15
101-201	0.15	0.12	0.13	0.2	0.12	0.15
101-202	0	0	0	0	0	0
101-203	0	0	0	0.33	0.04	0.07
101-204	0	0	0	0.33	0.04	0.07
101-205	0.17	0.12	0.14	0.26	0.16	0.2
101-206	1	1	1	0	1	0
101-207	0.32	0.2	0.25	0.24	0.08	0.12
101-208	0	0	0	0.33	0.04	0.07
101-209	0.17	0.14	0.15	0.17	0.14	0.15
101-210	0.34	0.2	0.25	0.22	0.08	0.12
101-222	0.33	0.5	0.39	0.67	0.47	0.55
101-223	0.25	0.26	0.26	0.13	0.07	0.09
101-224	0.23	0.16	0.19	0.24	0.08	0.12
101-225	0	0	0	0.33	0.04	0.07
101-230	0.33	0.2	0.25	0.28	0.11	0.16
101-231	0	0	0	0.33	0.04	0.07
101-237	0.31	0.47	0.37	0.67	0.47	0.55
101-238	0.25	0.26	0.26	0.13	0.07	0.09
101-249	0	0	0	0	0	0
101-251	0	0	0	0	0	0
101-252	0.04	0.03	0.03	0	0	0
101-258	0	0	0	0	0	0
101-259	0.02	0.03	0.03	0	0	0
101-301	0.12	0.2	0.15	0.12	0.2	0.15
101-302	0.4	0.26	0.31	0.58	0.23	0.33
101-303	0.02	0.1	0.04	0.02	0.1	0.04
101-304	0.02	0.04	0.03	0.05	0.07	0.06

IV. EXTENSION WITH FUZZY AGGREGATION OPERATORS

The subsumption degree is computed by combining $sd(l_s, l_S)$, $sd(R_s, R_S)$, and $sd(R_s, hypo_S)$. As shown in Equation 1, so far it is computed as a weighted sum. In this section, we will discuss alternative ways to do that.

Aggregation Operators (AOs) are mathematical functions that are used to combine different pieces of information [6]. Usually, given a domain $\mathtt{D}$ (for us, $\mathtt{D} = [0, 1]$), an AO of dimension n is a mapping $@ : \mathtt{D}^n \to \mathtt{D}$. Thus, an AO aggregates the values of n different criteria. Some examples of AOs are *maximum*, *minimum*, *order statistic*, *arithmetic mean*, *weighted sum* (or weighted mean), and *median*.

An important class of AOs are the *Ordered Weighted Averaging* (OWA) operators [7], a parameterized class of mean type AOs. Each OWA operator is parameterized with a vector of n weights $W = [w_1, \ldots, w_n]$ such that $w_i \in [0, 1]$ and $\sum_{i=1}^{n} w_i = 1$. Formally, an OWA operator of dimension n is an AO such that $@_W^{\text{owa}}(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i x_{\sigma(i)}$, where σ is a permutation of the values x_i such that $x_{\sigma(1)} \geq x_{\sigma(2)} \geq \cdots \geq x_{\sigma(n)}$, i.e., $x_{\sigma(i)}$ is the i-th largest of the values $x_1, \ldots, x_n$. This reordering step is a fundamental aspect of these operators: a weight w_i is not associated with a specific argument but with an ordered position of the aggregate. As a result, the OWA operator is non-linear. By choosing different weights, OWA operators can implement different AOs, such as arithmetic mean, k-th maximum, k-th minimum, median or order statistic, among others. However, it is worth to stress that weighted sum cannot be represented as an OWA operator.

To compute the subsumption degree, we argue that in some application domains, OWA can be more appropriate than a weighted sum because each of the weights is not directly associated to any of the aggregated values $sd(l_s, l_S)$, $sd(R_s, R_S)$, and $sd(R_s, hypo_S)$. For example, a vector of the form $W = [0.7, 0.2, 0.1]$ gives more importance to the highest of the values, so it is an optimistic computation of the subsumption degree. On the contrary, in the vector $[0.1, 0.2, 0.7]$ the lowest value has a greater contribution to the subsumption degree, so it is a more conservative decision.

To illustrate the usefulness of OWA, assume that we are aligning two ontologies written in different languages such as 101, in English, and 206, in French.[7] In several cases the similarity string metric will produce low values (terms in different values are expected to have different names, as it happens with the English 101:Book and the French 206:Livre), but in such case the lexical similarity is not significant, so we might not want to have it a high importance. However, if two terms happen to have a high lexical similarity despite of being expressed in different languages (for example, they both can have a Greek or Latin root, as in happens with the English 101:Monograph and the French 206:Monographie), then similarity is more significant. Thus, the importance is not always associated only to the attribute but also to its value.

One might consider instead a *t-norm* (a commutative, associative, and monotonic function $\otimes : [0, 1] \to [0, 1]$ with neutral element 1) or a *t-conorm* (a commutative, associative, and monotonic function $\oplus : [0, 1] \to [0, 1]$ with neutral element 0) [8]. The largest t-norm (the minimum) and the smallest t-conorm (maximum) correspond to the two extreme cases of OWA operators: $\min(x_1, \ldots, x_n) \leq @(x_1, \ldots, x_n) \leq \max(x_1, \ldots, x_n)$. The particular choice of the t-norm or t-conorm will depend on the application domain.

V. OTHER EXTENSIONS AND CONSIDERATIONS

A. *Use of Formal Concept Analysis*

It is worth mentioning some similarities and differences between our approach and Formal Concept Analysis (FCA) [9]. FCA tries to build a concept hierarchy or formal ontology (we will call it *FCA ontology*) from a collection of objects (or individuals) and the values of their properties. The intuitive idea is that if several objects share the same property values, there should be a general concept to which they belong. A class in the concept hierarchy corresponds to a set of objects sharing the same values for a set of properties, and each of its subclasses represent a subset of their objects, that is, objects sharing the same values for a subset of the properties.

In our case, the collection of objects are the ontology concepts from the ontologies to be aligned and their properties are the ontology roles. Rather than focusing on the precise values of the properties, we consider the existence of the attribute, i.e., if the classes *have* a property or not (that is, if the class is part of the domain or range of the property).

FCA is focused on the discovery of new hidden concepts rather than using existing concepts to build relationships

[7] For the sake of readability, we will shorten the ontology namespaces (for example, 101:Book denotes the concept Book from the 101 ontology).

between them as we do. Nevertheless, we can still use it in our scenario. The fact that a concept from the source ontology is subsumed by another one from the target ontology (or vice versa) according to the FCA ontology is coherent with having an alignment between them. Thus, we can use FCA in two different ways: *a priori*, as an initial step providing candidates for the alignments, or *a posteriori*, as a final step modifying the confidence of the computed alignments.

It should be noted that the candidates produced by the *a priori* way are not complete (because of the incomplete information in real-world ontologies) nor necessarily correct (because having more attributes is a not a sufficient condition).

For example, the concept 222:Deliverable is a subclass of 101:Deliverable. 101:Deliverable has 51 properties (it is the domain of 48 properties and the range of 3), and 222:Deliverable specializes it with 4 additional attributes (it is the domain of 222:chapter, 222:isPartOf, and 222:pages, and the range of 222:parts). Therefore, FCA identifies 222:Deliverable as more specific than 101:Deliverable, which supports the hypothesis of the subsumption relationship. However, we found many examples where the incomplete information in the aligned ontologies produces that the general concept has more properties than the subsumed one, so FCA would not produce this pair as a candidate. This is the case, e.g., of 222:Report with more properties than its subclass 101:TechReport.

This example assumed that two roles are the same if the have the same label. However, our approach improves this idea by considering a synonymy degree between the properties.

B. Use of Synonymy Relationships

It is very important to take into account the extensive work in discovery of synonymy relationships. A first possibility is to compute both a hyponymy degree (using our approach) and a synonymy degree (using any of the approaches in the literature), and to use the maximum of them to decide if two terms are synonyms or hyponyms (or none of them if both degrees are too low). However, it is important to ensure that both degrees are comparable, which requires some experiments to set the parameters and weights of both degrees so they are coherent with each other.

The use of lexical similarity in the discovery of hyponymy relationships is different from its use to find synonyms. The fact that two candidate terms have a similar name strongly supports that they are synonyms (even if it is not neither a sufficient or necessary condition). However, when looking for hyponyms the situation is different: two terms with a very similar name are less likely to be hyponyms than synonyms. However, two terms where the name of one of them is a substring of the name of the other one, are typically likely to be hyponyms: the longest name could be specializing the term and hence could be a hyponym.

C. Repeatability of the Evaluations

Now we will discuss some issues that are important to guarantee the repeatability of the evaluation results. Although using external third-party services or databases can be beneficial, it is important to make sure that the experiments are repeatable. For example, while *Wordnet* [10] offers a good control of versions and can be safely used, this is not always the case for other services. In fact, the availability of the external services can be severely compromised for limitations in its use, such as limited number of web searches, API restrictions, etc.

Besides, to compute precision and recall easily, it is desirable that the reference alignments do not include redundancies. This happens, e.g., in the OAEI oriented track 2009, where the fact that the concept 101:Collection is a subclass of the concept 304:Book appears twice. Special care is also needed to avoid counting correct subsumptions (e.g., 101:List is a subclass of rdf:List) but not included in the reference alignments due to the fact that they involve elements using different prefixes than the source and target ontologies. If we use a semantic reasoner to navigate through a concept hierarchy, we can get axioms about elements from the imported ontologies, with prefixes such as rdf, rdfs, foaf, etc.

VI. Related work

This section recaps some related work on the discovery of subsumption relationships in ontologies and the use of fuzzy logic or Formal Concept Analysis in ontology alignment.

A. Discovery of Subsumption Relationships

Most of the work in ontology alignment is focused on the discovery of synonymy relationships, and only a few works consider the discovery of subsumption relationships. Some of these previous works are based on the extraction of subsumption relationships on shared instances (such as [11] or [12]), whereas [4] deals with the extraction of hyponymy relationships at the schema level. Previous approaches extracting relationships at the schema level include the systems MOMIS [13], SCARLET [14], and Classification-based learning of Subsumption Relations (CSR) [15]. The alignments that MOMIS and SCARLET can find must already exist in third-party sources (Wordnet and other ontologies, respectively); [4] also exploits some external sources (such as Wordnet) but it can discover new relationships. Contrary to [4], CSR uses machine learning techniques so it requires a previous training step. The authors of CSR recognize that not all the ontologies are suitable for the training step.

B. Fuzzy Logic and Ontology Alignment

Whereas our approach use standard (crisp) ontologies, most of the previous work using fuzzy logic in ontology alignment/mapping considers *fuzzy ontologies* [16], [17], [18], [19], [20], [21]. Fuzzy ontologies have a different semantics [22], where concepts and properties are interpreted as fuzzy sets and fuzzy relations, respectively. While in our case the subsumption degree indicates the confidence of the system in the existence of such a relationship, in fuzzy ontologies it expresses to which extent the membership to the subsumed concept implies the membership to the superclass. Some exceptions considering crisp ontologies are [23], using fuzzy conceptual graphs to infer relantionships between instances, and [24], using fuzzy similarities to infer synonymy relationships, but we instead aim at discovering subsumptions between concepts.

C. Formal Concept Analysis and Ontology Alignment

While there are also some previous approaches using FCA in ontology alignment/mapping/merging [21], [25], [26], [27], none of them use FCA a posteriori as we suggested. Moreover, most of them do not combine FCA with other techniques such as label analysis or co-hyponym analysis as we do. There is an exception but it does not consider dynamic thresholds or fuzzy logic [28]. Furthermore, some of these works focus on the discovery of synonymy relationships rather than hyponymy relationships; in our opinion, since FCA produces a concept hierarchy of subsumption relationships, it actually seems much more natural to use it to build hyponymy relationships.

VII. Conclusions and future work

This paper has shown that using fuzzy logic makes it possible to improve a previous system for the automatic discovery of subsumption relationships between elements of classical ontologies. In particular, the use of fuzzy clustering is more stable, faster, and makes it easier the detection of cases where there are no subsumptions. Moreover, fuzzy aggregation operators offer a wide range of possibilities to combine the different measures used in our system. We have also discussed how to integrate existing reasearch on Formal concept Analysis and the discovery of synonym relationships.

Future work will include a revision of the parameters involved in the subsumption degrees (as they do not seem precise enough), an evaluation of the presented ideas that were not evaluated empirically (such as the use of OWA or FCA), and a study of the scalability for pairs of large ontologies.

Acknowledgment

We have been partially supported by the projects TIN2013-46238-C4-4-R, TIN2016-78011-C4-3-R (AEI/FEDER, UE), and DGA/FEDER. We would like to thank Enrique Solano-Bes for providing us with the source code used in [4], and Jesús Bermúdez for some helpful discussions.

References

[1] S. Staab and R. Studer, Eds., *Handbook on Ontologies*, ser. International Handbooks on Information Systems. Springer, 2004.

[2] M. Ehrig, *Ontology Alignment: Bridging the Semantic Gap*. Springer, 2006.

[3] C. Bobed, R. Yus, F. Bobillo, and E. Mena, "Semantic reasoning on mobile devices: Do androids dream of efficient reasoners?" *Journal of Web Semantics*, vol. 35, no. 4, pp. 167–183, 2015.

[4] R. Yus, E. Mena, and E. Solano-Bes, "Generic rules for the discovery of subsumption relationships based on ontological contexts," in *Proceedings of the 2015 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2015), Volume I*. IEEE, 2015, pp. 309–312.

[5] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, 1981.

[6] V. Torra and Y. Narukawa, *Modeling decisions - Information fusion and aggregation operators*. Springer, 2007.

[7] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decision making," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

[8] G. J. Klir and B. Yuan, *Fuzzy sets and fuzzy logic: theory and applications*. Prentice-Hall, Inc., 1995.

[9] B. Ganter and R. Wille, *Formal Concept Analysis*. Springer, 1999.

[10] G. A. Miller, "WordNet: A lexical database for English," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[11] D. Kang, J. Lu, B. Xu, P. Wang, and Y. Li, "A framework of checking subsumption relations between composite concepts in different ontologies," in *Proc. of the 9th Int. Conf. on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005), Part I*, ser. Lecture Notes in Computer Science, vol. 3681. Springer, 2005, pp. 953–959.

[12] R. Tournaire, J. Petit, M. Rousset, and A. Termier, "Discovery of probabilistic mappings between taxonomies: Principles and experiments," *Journal on Data Semantics*, vol. 15, pp. 66–101, 2011.

[13] D. Beneventano, S. Bergamaschi, S. Castano, A. Corni, R. Guidetti, G. Malvezzi, M. Melchiori, and M. Vincini, "Information integration: The MOMIS project demonstration," in *Proc. of the 26th Int. Conf. on Very Large Data Bases (VLDB 2000)*, 2000, pp. 611–614.

[14] M. Sabou, M. d'Aquin, and E. Motta, "SCARLET: semantic relation discovery by harvesting online ontologies," in *Proceedings of the 5th European Semantic Web Conference (ESWC 2008)*, ser. Lecture Notes in Computer Science, vol. 5021. Springer, 2008, pp. 854–858.

[15] V. Spiliopoulos, A. G. Valarakos, and G. A. Vouros, "CSR: discovering subsumption relations for the alignment of ontologies," in *Proceedings of the 5th European Semantic Web Conference (ESWC 2008)*, ser. Lecture Notes in Computer Science, vol. 5021. Springer, 2008, pp. 418–431.

[16] M. Bakillah and M. A. Mostafavi, "A fuzzy logic semantic mapping approach for fuzzy geospatial ontologies," in *Proceedings of the 5th International Conference on Advances in Semantic Processing (SEMAPRO 2011)*, 2011, pp. 21–28.

[17] A. Bahri, R. Bouaziz, and F. Gargouri, "Dealing with similarity relations in fuzzy ontologies," in *Proceedings of the 16th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007)*, 2007, pp. 1836–1841.

[18] A. Ferrara, D. Lorusso, G. Stamou, G. Stoilos, V. Tzouvaras, and T. Venetis, "Resolution of conflicts among ontology mappings: A fuzzy approach," in *Proc. of the 3rd Int. Workshop on Ontology Matching (OM 2008)*, ser. CEUR Workshop Proceedings, vol. 431, 2008.

[19] K. Todorov, C. Hudelot, A. Popescu, and P. Geibel, "Fuzzy ontology alignment using background knowledge," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 22, no. 1, pp. 75–112, 2014.

[20] B.-W. Xu, D.-Z. Kang, J.-J. Lu, Y.-H. Li, and J. Jiang, "Mapping fuzzy concepts between fuzzy ontologies," in *Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005), Part IV*, ser. Lecture Notes in Computer Science, vol. 3684. Springer, 2005, pp. 199–205.

[21] X. Xu, Y. Wu, and J. Chen, "Fuzzy FCA based ontology mapping," in *Proceedings of the 1st International Conference on Networking and Distributed Computing (ICNDC 2010)*. IEEE, 2010, pp. 181–185.

[22] F. Bobillo, M. Cerami, F. Esteva, À. García-Cerdaña, R. Peñaloza, and U. Straccia, "Fuzzy description logics," in *Handbook of Mathematical Fuzzy Logic Volume III*, ser. Studies in Logic, Mathematical Logic and Foundations. College Publications, 2015, vol. 58, ch. XVI, pp. 1105–1181.

[23] P. Buch, J. Dibie-Barthelemy, and L. Ibanescu, "Ontology mapping using fuzzy conceptual graphs and rules," in *Proceedings of the 16th International Conference on Conceptual Structures (ICCS 2008)*, vol. 354. CEUR Workshop Proceedings, 2008, pp. 17–.24.

[24] V. Cross and X. Hu, "Fuzzy set and semantic similarity in ontology alignment," in *Proceedings of the 21st IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2012)*. IEEE, 2012, pp. 1–8.

[25] K. X. S. de Souza and J. G. Davis, "Aligning ontologies and evaluating concept similarities," in *Proceedings of the 3rd International Conference on Ontologies, DataBases, and Applications of Semantics for Large Scale Information Systems (ODBASE 2004), Part II*, ser. Lecture Notes in Computer Science, vol. 3291. Springer, 2004, pp. 1012–1029.

[26] G. Stumme and A. Maedche, "FCA-MERGE: bottom-up merging of ontologies," in *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)*. Morgan Kaufmann, 2001, pp. 225–234.

[27] L. Guan-yu, L. Shu-peng, and Zhao-Yan, "Formal concept analysis based ontology merging method," in *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT 2010)*. IEEE, 2010, pp. 279–282.

[28] M. Zhao and S. Zhang, "Identifying and validating ontology mappings by formal concept analysis," in *Proceedings of the 11th International Workshop on Ontology Matching (OM 2016)*, ser. CEUR Workshop Proceedings, vol. 1766, 2016, pp. 61–72.