Generic Rules for the Discovery of Subsumption Relationships based on Ontological Contexts

Roberto Yus, Eduardo Mena, and Enrique Solano-Bes
University of Zaragoza
Maria de Luna 1, 50018 Zaragoza, Spain
Email: {ryus,emena}@unizar.es, esolanobes@gmail.com

***Abstract*—Structured data extracted from the Web is highly heterogeneous due to its disparate origins and nature. There exist some techniques to integrate this information based on the extraction of synonymy relationships among the different entities involved. However, synonymy is a very strict and therefore uncommon relationship. We present a novel approach for the discovery of subsumption relationships among concepts from different ontologies. Our approach is based on the use of generic rules, designed to capture the existence of a subsumption relationship, considering features of the ontological context of concepts (i.e., labels, roles, and hierarchical relationships).**

***Keywords*-Schema mapping/matching, ontology alignment, subsumption relationships extraction**

I. Introduction

There has been a considerable amount of work in the ontology alignment [2] area. Nevertheless, most works are focused on the alignment through the extraction of synonymy relationships [4]. However, synonymy is a very strict relationship that implies, in fact, that the two entities have the *same meaning*. In the real world it is more common to find terms that are similar but not exactly the same (e.g., one of the terms could be more general, it could *subsume* the other term). There are only a few works focused on discovering subsumption relationships and they are based on: instances in the ontologies [3] (but not all the ontologies contain enough instances for that); external sources where the relationships could be defined [1], [5] (but sometimes the relationships are not defined anywhere); and classification methods [6] (which depend on the training data).

We present a novel approach to the discovery of subsumption relationships between concepts from different ontologies. The main contributions of our approach are: 1) To our knowledge, is the first one defining generic rules to discover subsumption relationships among concepts from ontologies; 2) we present a formula, based on the rules, to compute the subsumption degree between two concepts by leveraging their ontological context (such as labels, roles, and potential cohyponyms); and 3) we introduce a phase to discard relationships, with respect to their subsumption degree, based on the computation of an automatic threshold using clustering techniques.

In addition, we present a preliminary experimental evaluation with different third-party ontologies extracted from the Web and the standard dataset from the OAEI (Ontology Alignment Evaluation Initiative). The good precision and recall achieved in the experiments, some of them with challenging ontologies for our method, show that our approach is promising for other ontologies too.

II. Architecture of the System

We propose the following steps to discover subsumption relationships (see Fig. 1):

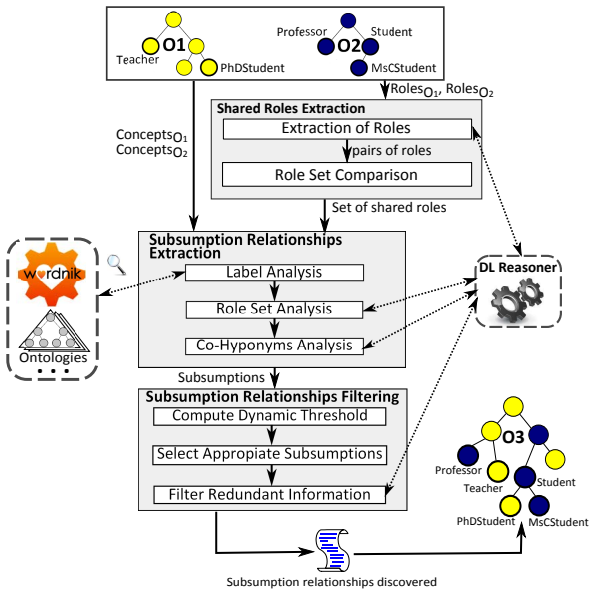

Figure 1. Main steps proposed to discover subsumption relationships.

1) *Shared roles extraction:* The set of roles (implicitly and explicitly) belonging to each concept are extracted and compared to extract the list of shared roles between concepts.
2) *Subsumption relationships extraction:* The subsumption degree among the concepts is computed by using their ontological context (i.e., labels, shared roles, and potential co-hyponyms).

3) *Subsumption relationships filtering:* The less probable relationships extracted are filtered out using a dynamic threshold to return a mapping file with the most probable subsumption relationships.

In the following we detail the last two previous steps.

III. Subsumption Relationships Extraction

Our goal is to discover the possible subsumption relationships among the concepts[1] of two ontologies using their ontological contexts by considering $C_s \sqsubseteq C_S,\ \forall\ C_s \in O_1, C_S \in O_2$. For this, our approach computes a *subsumption degree* d that indicates the confidence of the system on the existence of such a relationship as:

$$w_l * sd(l_s, l_S) + w_r * sd(R_s, R_S) + w_{ch} * sd(R_s, hypo_S)$$

where the subsumption degree of the labels of the concepts (l_x) is computed combining information about their relationships from third-party lexical databases and their similarity string metric [7]. The subsumption degree of their roles (where $R_x = \{r, C_x \in domain(r)\}$) and potential co-hyponyms of C_s are explained in Section III-A and Section III-B, respectively. Notice that some of the roles in R_s and R_S are inherited from the hypernyms of their concepts (i.e., if $C_s \sqsubseteq C' \wedge C' \in domain(r) \rightarrow C_s \in domain(r)$) and some of the roles in these sets are not explicitly asserted but inferred using a Description Logics (DL) reasoner. Also, a role r is "shared" by the concepts C_s and C_S if $r \in R_s$ and $r \in R_S$. Finally, w_l, w_r, and w_{ch} are some weights assigned to each factor with $w_l + w_r + w_{ch} = 1$.

A. Role Set Analysis

The roles of the concepts can be used to find "hints" related to the features that every concept, C_s, subsumed by another concept, C_S, presents:

Statement 1: C_s must have all the roles of C_S since a concept inherits all roles of its subsumer.

Statement 2: C_s should have more roles than C_S (i.e., it should be more specialized).

In the ideal case, both statements should be followed; however, in a real scenario different situations may happen:

Statement 3: It is possible that C_s does not have all the roles of C_S although it is true that $C_s \sqsubseteq C_S$.

Statement 4: We could find concepts which do not have any role characteristic enough to discover the semantics of a concept (all its roles are inherited).

Given two concepts and their set of roles we can define a formula that takes the previous statements into account to obtain their subsumption degree. The graphical representation of the desired subsumption degree with respect to the number of roles that the two concepts share, considering the number of roles of the subsumer concept, could be similar to the graph in Fig. 2. The important aspect of the graph in Fig. 2 is not the specific values shown (which simply correspond to our prototype) but that it models the following generic rules that capture the existence of subsumption relationships:

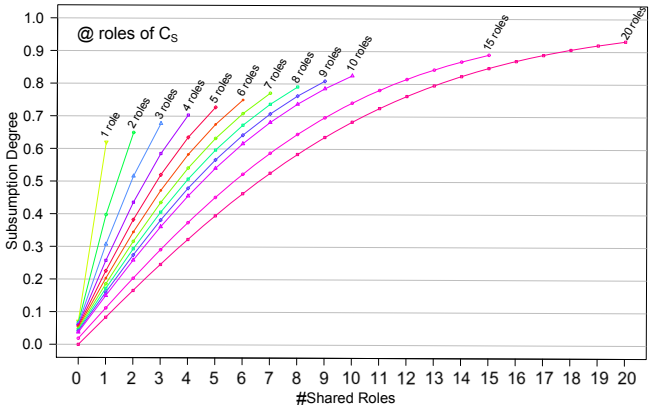

Figure 2. Subsumption degree (Y-axis) between two concepts, C_s and C_S, depending on the number of roles of C_S (denoted by the different curves) and their shared roles (X-axis).

Rule 1: The higher the percentage of roles of C_S that C_s has, the greater the subsumption degree.

According to Statement 1, a subsumed concept should inherit 100% of the roles of its subsumers, although sometimes this percentage is less (Statement 3). If C_s^1 and C_s^2 share 40% and 80% of C_S roles, respectively, then $d(C_s^2, C_S)$ should be greater than $d(C_s^1, C_S)$.

Rule 2: The higher the number of shared roles, the greater the subsumption degree.

If C_s^1 shares one role with C_S^1 (suppose that is 50% of C_S^1's roles in this case) and C_s^2 shares six roles with C_S^2 (which is also 50%), then $d(C_s^2, C_S^2)$ should be greater than $d(C_s^1, C_S^1)$. I.e., according to the Duck Test: "If it looks like a duck, swims like a duck, and quacks like a duck, then it probably is a duck."; however, if it only swims like a duck, the probability of being a duck is lower.

In the following we present three rules that detail Rule 2 in case of C_s sharing all, none, and n of C_S roles:

Rule 2.1: If C_s shares all the roles *of C_S (the ideal case according to Statement 1), the higher the number of roles C_S has, the higher the subsumption degree between them.*

If C_s^1 shares one role with C_S^1 (that is 100% as C_S^1 only has one role) and C_s^2 shares six roles with C_S^2 (which is also 100% as C_S^2 has six roles), then $d(C_s^2, C_S^2)$ should be greater than $d(C_s^1, C_S^1)$. Again, the Duck Test applies. In addition, the difference between these maximum values cannot be linear according to the number of C_S roles as, for any number of roles, the subsumption degree function should score between 0 and 1 always. Hence, the subsumption degree should grow slower for a high number of shared roles as beyond a certain amount of shared roles the subsumption degree should be close to 1.

[1]In Description Logic subsumption exists also among roles, however we only focus on subsumption among concepts which is more common.

Rule 2.2: If C_s shares no role with C_S (this could happen according to Statement 3), the higher the number of roles C_S has, the lower the subsumption degree between them.

If C_s^1 shares no role with C_S^1 (C_S^1 has one role in this case) and C_s^2 shares no role with C_S^2 (C_S^2 has six roles in this case), then $d(C_s^2, C_S^2)$ should be lower than $d(C_s^1, C_S^1)$. According to what we could call 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. However, if it only does not swim like a duck, the probability of being a duck is higher. Also, subsumption degrees for all these minimum values have to be lower than situations where C_s shares one or more roles with C_S, which is always better than not sharing any role at all.

Rule 2.3: If C_s shares n of the m roles of C_S ($0 < n < m$), the higher the number of shared roles, the greater the subsumption degree.

Intuitively, we could think that the number of non-shared roles, $m - n$, could "neutralize" the number of shared roles. However, according to the spirit of Rule 2, we believe that the greater number of shared roles, the more hints of C_s being subsumed by C_S. We could call it the *Weak Duck Test*: if it looks like a duck and quacks like a duck, then it is probably a kind of duck, although we are not sure that it swims like a duck. In other words, we advocate that shared roles score more than what non-shared roles penalize the subsumption degree.

The number of characteristic roles of C_s that are not inherited from C_S (Statement 2) do not affect the graph: To determine whether C_s is subsumed by C_S or not only shared roles are taken into account. For example, if the concept $Person$ has four roles and the concept $PhDStudent$ has these four and five extra roles, the latter would be a subconcept of $Person$ regardless of its extra roles (the extra roles could, at most, indicate that $PhDStudent$ might be subsumed by another concept). Also, not all the roles should have the same importance at the time of computing the subsumption degree. In general, the more concepts share the role the less important it is for extracting a subsumption relationship (Statement 4).

To model these rules, and so the trend in Fig. 2, we used a logistic function that obtains the subsumption degree:

$$dRoles(C_s, C_S) = 2 * \left(\frac{1}{1 + e^{-a * f(C_s, C_S)}} - 0.5 \right) \quad (1)$$

where the function $f(C_s, C_S)$ computes the subsumption degree between concepts C_s and C_S as:

$$f(C_s, C_S) = w_{sh} * \frac{sh(C_s, C_S)}{|C_S|} + w_{diff} * \frac{diff(C_s, C_S) - min}{max - min} \quad (2)$$

$$diff(C_s, C_S) = \frac{1}{\alpha} * sh(C_s, C_S) - |C_S| \quad (3)$$

where C_s and C_S represent their sets of roles and the two terms in Formula 2 are: 1) the percentage of C_S's roles that C_s has (Rule 1), being $sh(C_s, C_S)$ the number of shared roles and $|C_S|$ the number C_S's roles; 2) the number of C_S's roles that C_s has (Rule 2) with respect to the rest of the ontology, being $diff(C_s, C_S)$ the difference between the number of shared and non-shared roles, α is a constant that models the importance of the number of shared and non-shared roles, and max and min are the maximum and minimum number of concepts that share the same role in the source ontology and are used to normalize the term. Also, w_{sh} and w_{diff} are used to adjust the importance of the shared roles and non-shared roles, respectively (w_{sh} should be greater according to Rule 2.3).

In addition, a modifier is applied to each role r when computing $sh(C_s, C_S)$ that reduces the subsumption degree of concepts sharing very common roles in the ontology:

$$uniqueness(r) = 1 - k * \left(\frac{\#domains}{\#maxDomains} \right) \quad (4)$$

being *#domains* is the number of concepts in the ontology that have the role, except C and the concepts subsumed by C, and *#maxDomains* is the maximum number of concepts that may have r as part of their definition. For example, imagine that an ontology has only three roles r_1, r_2, and r_3 which have 3, 10, and 5 concepts as their domains, respectively, then $\#maxDomains = MAX(3, 10, 5)$.

B. Co-hyponyms Analysis

In general, two co-hyponyms concepts, C_s and $C_{s'}$ such that $(C_s \sqsubseteq C_S) \wedge (C_{s'} \sqsubseteq C_S)$, will share the roles of C_S (remember Statement 1) but they could share other roles too, especially in the absence of intermediate concepts in the ontology. For example, consider a "missing concept" C_m such that $(C_{s'} \sqsubseteq C_m) \wedge (C_s \sqsubseteq C_m) \wedge (C_m \sqsubseteq C_S)$. In this scenario C_s and $C_{s'}$ will share some roles that C_S does not have, the roles inherited from C_m.

To compute the similarity of the concept C_s and the subsumed concepts of C_S, we compare the sets of roles of each subsumed concept, C_i, with the set of roles of C_s individually. This measure is calculated by computing the average between two terms: 1) the amount of roles shared by C_s and the subsumed concept C_i, with respect to the number of roles of C_s, and 2) the amount of roles shared with respect to the number of roles of C_i. Once the system has the similarity degree for each subsumed concept then it calculates the average similarity of all co-hyponyms.

$$dCohyp(C_s, C_S) = \frac{\sum_{C_S^{childs}}^{C_i} \left(\frac{\frac{sh(C_s, C_i)}{|C_s|} + \frac{sh(C_s, C_i)}{|C_i|}}{2} \right)}{\left| C_S^{childs} \right|} \quad (5)$$

being C_S^{childs} the set of subsumees of C_S (i.e., the potential co-hyponyms of C_s) and $\left| C_S^{childs} \right|$ the number of elements in the set, C_i is the set of roles of the concept C_i and $|C_i|$ and $|C_s|$ are the number of roles of C_i and C_s, respectively.

IV. Subsumption Relationships Filtering

Our approach computes the subsumption degree among all the pairs of concepts from the two ontologies. However, there are three major groups of relationships discovered according to their subsumption degree: very probable as the degree is high, clearly unrelated concepts as the degree is very low, and questionable relationships with a neither high nor low degree. Our approach automatically discards those values that are not probable enough by using three filters to:

Discard subsumptions under a (dynamic) threshold. The trend of the subsumption degrees computed depends on the ontology and thus, the threshold to filter out less probable relationships should be dynamic. Instead of using a classifier that would require training on the domain, we propose using a clustering algorithm that automatically divides relationships into the three groups explained before.

Select between hypernymy and hyponymy. Our approach obtains the subsumption degree for all the possible combinations of concepts and so, values for both $C_s \sqsubseteq C_S$ and $C_S \sqsubseteq C_s$ are computed. Selecting both relationships, even if the degree is low, will create a synonymy relationship between the concepts and that is out of the scope of the approach as it would require further analysis. Therefore, this filter discards the relationship with the lower degree.

Discard redundant relationships. This filter selects the relationship with the higher degree for a given concept from all the potentially redundant relationships discovered.

The final list of subsumption relationships and the original axioms can be materialized to create an integrated ontology. Some definitions of concepts can be contradictory so we advocate inserting the subsumption axioms discovered one by one, in descending subsumption degree order, and using a DL reasoner to remove an inserted axiom if the ontology is inconsistent after its insertion.

V. Experimental Evaluation

To test our approach we developed a prototype and carried out several preliminary tests with different ontologies[2]: two well-defined ontologies as an example of an "ideal" scenario (O_1, O_2); some ontologies from the OAEI 2009 dataset (*101*, *222*, *223*, *304*); and some challenging real-world ontologies extracted from the Web (*univCs*, *unicBench*, *confOf*, *sigkdd*, *conference*). Table I shows precision and recall of the extracted subsumption relationships for these tests.

Even in extreme scenarios for our approach where some roles do not have a domain defined or not many roles are defined, our prototype obtained fairly good results with an average F-measure of 0.65. Take into account that in these situations even experts would have problems to discover relationships manually. It could be interesting to study the application of a preprocessing phase, to refine the domain and range of roles, and the use of a more sophisticated clustering algorithm for the filtering phase, our prototype used *k-means*. In Table I we show how manually adjusting the automatic threshold improved the results for some tests (*101-223'*, *101-304'*, *confOf-sigkdd'*, *confOf-conference'*).

[2]More information about the experiments and the ontologies used can be found at http://sid.cps.unizar.es/SubsumptionExtraction

Test	#relations	Precision	Recall	F-measure
O_1-O_2	26	0.96	0.89	0.93
101-222	32	0.96	0.73	0.83
101-223	82	0.65	0.41	0.50
101-223'	84	0.72	0.63	0.67
101-304	78	0.38	0.47	0.42
101-304'	48	0.83	0.44	0.58
univCs-unicBench	74	0.77	0.63	0.69
confOf-sigkdd	24	0.58	0.67	0.62
confOf-sigkdd'	21	0.67	0.67	0.67
confOf-conference	41	0.61	0.58	0.60
confOf-conference'	27	0.74	0.47	0.57

Table I
Precision and recall in our tests.

As a summary, these experiments show that our approach would be able to achieve good results with well-defined ontologies (with an average F-measure of 0.88 for the tested ontologies), and promising results with ontologies that caused problems to other systems (average F-measure of 0.64). The specific values achieved, although good, depend on the ontologies considered but these experiments show that our approach looks promising for being the first to discover subsumption relationships using generic rules.

Acknowledgments. Research work supported by CICYT project TIN2013-46238-C4-4-R and DGA-FSE. We thank Jorge Bernad, Jorge Gracia, and Jesus Bermudez for their comments.

References

[1] D. Beneventano, S. Bergamaschi, S. Castano, A. Corni, R. Guidetti, G. Malvezzi, M. Melchiori, and M. Vincini. Information integration: the MOMIS project demonstration. In *26th Int. Conf. on Very Large Data Bases*, 2000.

[2] J. Euzenat and P. Shvaiko. *Ontology matching*. Springer, 2007.

[3] D. Kang, J. Lu, B. Xu, P. Wang, and Y. Li. A framework of checking subsumption relations between composite concepts in different ontologies. In *9th Int. Conf. on Knowledge-Based Intelligent Information and Engineering Systems*, 2005.

[4] S. Pavel and J. Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. on Knowl. and Data Eng.*, 25(1), 2013.

[5] M. Sabou, M. d'Aquin, and E. Motta. SCARLET: semantic relation discovery by harvesting online ontologies. In *5th European Semantic Web Conf.*, 2008.

[6] V. Spiliopoulos, A. G. Valarakos, and G. A. Vouros. CSR: discovering subsumption relations for the alignment of ontologies. In *5th European Semantic Web Conf.*, 2008.

[7] G. Stoilos, G. Stamou, and S. Kollias. A string metric for ontology alignment. In *4th Int. Semantic Web Conf.*, 2005.