Do competency questions for alignment help fostering complex correspondences?

Elodie Thiéblin

Institut de Recherche en Informatique de Toulouse, France
elodie.thieblin@irit.fr

Abstract. The Linked Open Data is composed of *linked* knowledge bases. Most of these links are still limited to simple correspondences. As a complement, complex correspondences bring more expressiveness to bridge the heterogeneity of knowledge bases. Finding correspondences (simple or complex) is the purpose of ontology matching. Existing matching solutions mainly focus on establishing as many correspondences as possible given two knowledge bases. On the one hand, this has the effect of neglecting the user needs. On the other hand, when dealing with large knowledge bases, this may impact the performance of the matching task. In response to this observation, we introduce Competency Questions for Alignment (CQAs) to express the needs of a user with respect an ontology alignment. We present our work on how CQAs can help ontology matching, and in particular complex matching.

Keywords: competency questions, ontology matching, complex alignment

1 Problem definition

Ontology matching is an essential task for the management of the semantic heterogeneity in open environments [6]. The matching process aims at generating a set of correspondences (i.e., an alignment) between the entities of different ontologies. A distinction is made between simple alignments, which contain only atomic entity to atomic entity correspondences, and complex alignments, with at least a complex correspondence. Complex correspondences involve logical constructors (e.g., property restriction as in $o_1{:}AcceptedPaper \equiv \exists\ o_2{:}hasDecision.\{o_2{:}accept\}$) or transformation functions of literal values (e.g., string concatenation). These correspondences are more expressive than simple correspondences and come as their complement.

Despite the variety of matching approaches, most of them aim at fully aligning two ontologies, i.e., the output alignment aims at fully covering the common scope of the two ontologies. However, a user may not need as much coverage as he or she may be interested by only a part of the ontology scope. Moreover, when reducing the scope of the ontologies, the matching task can be performed more efficiently and even allow for on-the-fly ontology matching [10], in particular when dealing with large knowledge bases. One could argue that the matching task can be performed offline. However, even offline, when dealing with large ontologies as in the LargeBio track of the OAEI [1], the scale is an issue and few

systems can cope with it. The scale becomes even more problematic for complex matching where the number of possible correspondences is not $O(mn)$ as for simple matching, m and n being the number of entities from the source and target ontology, but worst than $O(2^{mn})$.

In order to address these observations, we define the notion of Competency Questions for Alignment (CQAs) to express the needs of a user with respect to the matching task. The CQAs represent the scope of the ontologies that the alignment should cover. This notion is inspired from the ontology authoring field, where competency questions have been introduced as *ontology's requirements in the form of questions the ontology must be able to answer* [7, 15, 17]. Our hypothesis is that CQAs can help ontology matching especially in the case of complex ontology matching. Indeed, focusing on the user needs may reduce the matching space and by consequence improve the matching performance. In this thesis, we aim at answering the general research question *Do CQAs help the fostering of complex correspondences?* This question can be broken down into *Can CQAs improve the quality of the generated alignments?* and *Can CQAs improve the run-time performance of complex ontology matching systems?*

This paper presents a state of the art of the problem (§2), the core of our proposition (§3), the methodology we plan to follow to answer the research questions (§4), the preliminary results obtained during my first two years of Ph.D. thesis (§5), and finally, a discussion and our short-term perspectives (§6).

2 State of the art

We distinguish two main aspects in our proposition. The first one is the involvement of the user in the matching process. The second one is the problem of complex ontology matching. We end this section with a discussion on how our approach is different from those in the literature.

User involvement in ontology matching A user may intervene in a matching process to express his or her needs. User involvement can be performed at different stages of the process: before, during or after. We make a distinction between three types of implication: user knowledge needs, user requirements and user validation. Most of the existing matching approaches involving the user focus on user validation.

User knowledge need The user knowledge needs express the expected knowledge content of an alignment: its scope. With regards to user specification of the alignments, the closest definition of user knowledge need is given in [10]. This paper presents a matching system which generates on-the-fly simple ontology alignments to cross query multiple knowledge bases. The queries used in the process are not exactly CQAs as we define them (see §3), but define *de facto* the content of the expected alignment.

User requirements User requirements are the specifications to the alignment (and the matching process). These specifications are not about the content of the alignment but about its properties. Few guidelines in the literature are given

to characterise an alignment and/or the matching process. The NeOn methodology [5] characterises both alignment and matching process through a set of questions: i) is matching performed under time constraints? ii) has matching to be performed automatically? iii) must the alignment be correct? complete? and iv) what type of operation (merging, query, etc.) is to be performed? Through these questions, qualitative and applicative characteristics of an alignment and the matching process are defined. However, they do not help specifying the knowledge the alignment should cover, i.e. its scope.
User validation User validation [4] helps the fostering of correct correspondences by providing partial alignments, correspondence validation, or correspondence completion as input to the matching process. However, it does not help define the scope of the alignment.

Competency questions In order to formalise the knowledge needs of an ontology, competency questions have been introduced in ontology authoring as *ontology's requirements in the form of questions the ontology must be able to answer* [7]. In [17], a competency question (CQ) in natural language can be expressed and translated into a SPARQL query. The authors define a set of characteristics to analyse competency questions (question type, element visibility, question polarity, predicate arity, modifier, domain independent element) based on both the natural language question and its associated SPARQL query. The work of [17] was corroborated by a recent study [3] on how users' interpretation of the CQs match the CQs author's intentions.

Complex matching approaches We can observe a growing interest in complex matching in the literature. These approaches involve different matching techniques relying on templates of correspondences (called patterns) and/or instance evidence. The approaches in [18, 19] apply a set of matching conditions (label similarity, datatype compatibility, etc.) to detect correspondences that fit certain patterns. The approach of [20] uses the linguistic frames defined in FrameBase to find correspondences between object properties and the frames. KAOM [9] relies on *knowledge rules* which can be interpreted as probable axioms. The approaches in [13, 14, 25] use statistical information based on the linked instances to find correspondences fitting a given pattern. The approach in [12] uses genetic programming on instances to find correspondences with value transformation functions between two knowledge bases. The one in [16] uses a path-finding algorithm to find correspondences between two knowledge bases with common instances. The one in [8] iteratively constructs correspondences based on the information gain from matched instances between the two knowledge-bases.

Discussion Comparing our proposal to those describe above, the approaches which involve the user (mostly for validation [2, 11]), or for user knowledge need expression [10]) do not deal with complex correspondences. On the other hand, none of the complex approaches involve the user before or during the matching process. Like the ones in [8, 13, 14, 25, 16], we rely on the hypothesis that the knowledge bases contain common instances. Furthermore, as for the matching processing in general, in particular for the complex matching approaches in [18,

19], we rely on the hypothesis that the ontologies in the knowledge base have a relevant lexical layer. Differently from [18, 19, 25, 14, 13], our approach does not rely on correspondence patterns. As far as we know, competency questions have not been adapted nor used for ontology matching.

3 Proposed approach

In this section, first we give a definition of CQAs with their characteristics. Then, we present our complex matching approach based on CQAs.

CQA definition A Competency Question for Alignment (CQA) can be defined as a Competency Question (CQ) that needs to be satisfied over two or more ontologies. Therefore, an alignment is needed. CQAs can not be used for Ontology Authoring whereas CQs can be. Hence, the scope of a CQA is limited by the intersection of its source and target ontologies' scopes. Another difference is that the maximal and ideal alignment's scope is not known *a priori* (as it is the purpose of the alignment). The characteristics defined by [17] for ontology authoring are applicable CQAs except the *predicate arity* which depends on the associated SPARQL query. Indeed, a CQA has not only one but as many associated SPARQL queries as ontologies that it should cover. For example, the CQA "What are the accepted papers ?" can be represented by `SELECT ?x WHERE {?x a o1:AcceptedPaper.}` in which there is only a unary predicate (`o1:AcceptedPaper`) with only explicit elements or by `SELECT ?x WHERE{?x a o2:Paper. ?x o2:hasDecision o2:accept.}` in which `o2:hasDecision` is a binary predicate and an implicit element of the query. We chose to adapt only the definition of predicate arity for the CQA into **question arity**. The **question arity** represents the arity of the expected answers to a CQA.

- A *unary* question expects a set of instances or values, e.g., "What are the accepted papers?" *(paper1), (paper2).*
- A *binary* question expects a set of instances or value pairs, e.g., "Who is the reviewer of a paper?" *(reviewer1, paper1), (reviewer1, paper2).*
- A *n-ary* question expects a tuple of size 3 or more, e.g., "What is the decision of a paper given by a reviewer?" *(paper1, reviewer1, accept), (paper3, reviewer2, reject).*

Complex matching approach proposition The proposed approach takes as input a set of CQAs in the form of SPARQL queries over the source ontology. It requires that the source and target ontologies have an $\mathcal{A}box$ with at least a common instance. The answer to each input query is a set of instances, which are matched with those of a knowledge base described by the target ontology. The matching is performed by finding the lexically similar surroundings of the target instances. CQAs for the approach are limited to *unary* questions (class expressions, set of instances expected), of *select* type, positive polarity and no modifier. The choice of the *select* question type, comes from the fact that *binary* and *counting* questions have a corresponding *select* question. With regards to the question polarity, a negative question implies that a "positive" information

is being negated, therefore, the questions can be limited to positive polarity only. We make the assumption that the user knows the source ontology and is able to write each CQA into a SPARQL on the source ontology. The approach is articulated in 11 steps, as depicted in Figure 1:

1. Extract source DL formula e_s from SPARQL CQA (e.g., o_1*:AcceptedPaper*)
2. Extract lexical information from the CQA, L_s set labels of atoms from the DL formula (e.g., "accepted paper")
3. Extract source instances $inst_s$ (e.g., o_1*:paper1*)
4. Find equivalent or similar (same label) target instances $inst_t$ to the source instances $inst_s$ (e.g. o_1*:paper1* $\sim$ o_2*:paper3*)
5. Retrieve the description of target instances: the set of triples in which the target instances appear as well as the object/subject type of the triple (e.g. in DL, the description of o_2*:paper3* would be $\langle$*(*o_2*:paper3,*o_2*:accept):*o_2*:hasDecision ;* o_2*:accept:*o_2*:Decision*$\rangle$; $\langle$*(*o_2*:reviewer1,*o_2*:paper3):*o_2*:reviewerOf ;* o_2*:reviewer1:*o_2*:Reviewer*$\rangle$)
6. For each triple, retrieve L_t labels of entities (e.g., o_2*:hasDecision* $\rightarrow$ "decision", o_2*:accept* $\rightarrow$ "accept", o_2*:Decision* $\rightarrow$ "decision")
7. Compare L_s and L_t using a string comparison metric (e.g., Levenshtein distance with a threshold)
8. Keep the triples with the summed similarity of their labels above a threshold τ. Keep the object/subject type if its similarity is better than the one of the object/subject (e.g. sim(o_2*:accept*, L_s) $>$ sim(o_2*:Decision*,L_s) so we only keep o_2*:accept* in the triple)
9. Express the triple into a DL formula (e.g., $\exists$ o_2*:hasDecision.*$\{o_2$*:accept*$\}$)
10. Aggregate the formulas into an explicit or implicit form. If two DL formulas have a common atom in their right member (target member).
11. Put the DL formulae e_s and e_t together in a correspondence (e.g., o_1*:AcceptedPaper* $\equiv$ $\exists$ o_2*:hasDecision.*$\{o_2$*:accept*$\}$) and express this correspondence in EDOAL

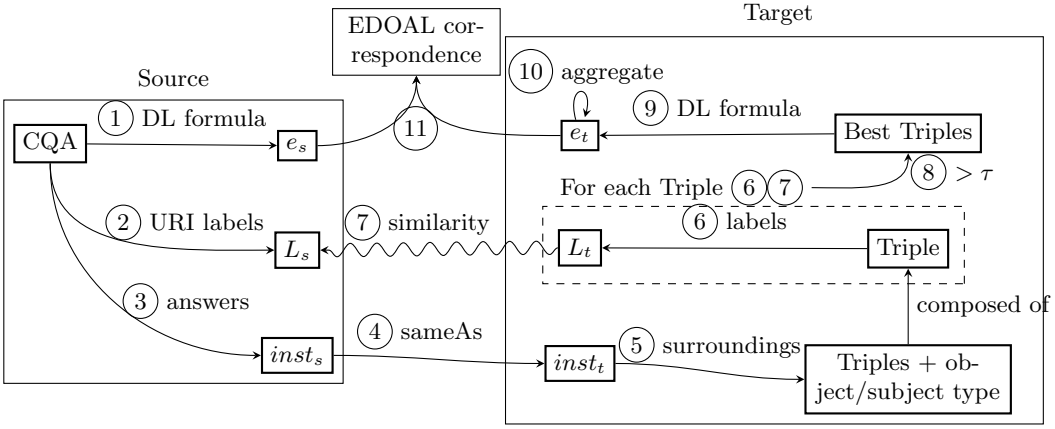

Fig. 1: Schema of the general approach.

4 Methodology

This thesis aims at addressing the following research questions:
What is the impact of CQAs on the proposed matching approach? We plan on comparing the output of the system when given manually created CQAs, versus a version of the tool based on automatically generated queries instead of CQAs. These two versions will be compared following the methodology described in the following research questions.
Can CQAs improve the quality of generated alignments? We plan on comparing state-of-the-art matching approaches with our approach in terms of manually assessed quality (precision). Currently, automatic evaluation of ontology alignment are only available for simple alignments. The automatic evaluation of complex alignments is out of the scope of this thesis.
Can CQAs improve the run-time performance of complex ontology matching systems? We plan on comparing the run-time of state-of-the-art complex matching systems with our system, in particular on large knowledge bases.
What is the impact of the CQA on the type of output correspondence? This research question aims at assessing if the use of CQAs makes an alignment more complex than it could be. To answer this question, we plan on comparing the output of our approach with a gold standard alignment having the same scope and count the number of complex correspondences which could be decomposed into simple correspondences.

5 Preliminary results

A first version of the approach has been implemented. This version uses a Levenshtein distance with a threshold as similarity metric and only deals with unary CQAs. It has been evaluated on large plant taxonomy knowledge bases: Agronomic Taxon, AgroVoc, TaxRef-LD and DBpedia [23]. For this baseline evaluation, CQAs have been manually generated by experts. The overall precision obtained was about 32.8% (44/134), while 83.4% (20/24) of the CQAs lead to the discovery of semantically equivalent correspondences. The correspondences found were mostly relevant and few CQAs lead to no correct correspondence. However, these results have not yet been compared to state-of-the-art systems. The next step is to propose and implement a matching process for binary CQAs. Even if the matching process will also be based on competency questions, it will consider pairs of instances and the approach will imply a path-finding phase between the matched instances. In order to overcome the lack of complex benchmarking datasets, we have been working on the first complex track of the OAEI[1] [21]. As these datasets do not include CQAs, an automatic generator of queries has been implemented in order to automatically evaluate our approach. This generator creates a query for each class of an ontology populated with at least one instance. It also generates property-value pairs as unary queries. The version of the system with the query generator has been evaluated in this year's OAEI

[1] `http://oaei.ontologymatching.org/2018/complex/`

campaign. Regarding the future evaluation of our approach, we are currently populating the conference dataset ontologies [26] with more or less common instances. A reference alignment was proposed and detailed in [22]. We plan on proposing a benchmark for complex alignment evaluation based on the populated Conference dataset and on a set of queries to be rewritten for these ontologies. The precision and recall could then be measured over the gold standard queries and the ones returned by the queries rewritten from the evaluated alignment. The evaluation queries can be used as CQAs for our matching approach.

6 Discussion

Taking into account the knowledge needs of the user in the matching process is novel in the field. We propose to express these needs using CQAs. However, the creation of the CQAs as SPARQL queries implies that the user knows the source ontology. As for [10], we believe that our system would be suited for on-the-fly ontology matching to cross query heterogeneous databases. We list a non-exhaustive set of perspectives for this work. The use of CQA for ontology matching opens new perspectives such as ontology matching with natural language to ontology mapping techniques [24] over multiple ontologies. For now, the user involvement happens *before* the matching process. One could think of an interactive matching system helping the user reduce the scope of the alignment *during* the matching phase. Moreover, an evaluation of the user involvement in our approach would interesting. The instance matching phase of the system is rather naive (existing links and exact label match) but we do not plan on going further in that direction. The use of state-of-the-art instance matching systems may be considered in the future. We are aware of the limitations of the approach, especially because all aligned knowledge bases must contain at least one common instance. A future direction is to propose a system which would not need any instance.

Acknowledgements I would like to thank my Ph.D. advisors Cassia Trojahn and Ollivier Haemmerlé from IRIT; Catherine Roussey and Nathalie Hernandez for their contribution on the Agronomic dataset.

References

1. Achichi, M., Cheatham, M., Dragisic, Z., Euzenat, J., Faria, D., Ferrara, et al.: Results of the ontology alignment evaluation initiative 2017. In: Ontology Matching Workshop. pp. 61–113. No commercial editor. (2017)
2. Cruz, I.F., Antonelli, F.P., Stroe, C.: Agreementmaker: efficient matching for large real-world schemas and ontologies. Proceedings of the VLDB Endowment (2009)
3. Dennis, M., Van Deemter, K., Dell'Aglio, D., Pan, J.Z.: Computing authoring tests from competency questions: Experimental validation. In: International Semantic Web Conference. pp. 243–259. Springer (2017)
4. Dragisic, Z., Ivanova, V., Lambrix, P., Faria, D., Jiménez-Ruiz, E., Pesquita, C.: User Validation in Ontology Alignment. In: Groth, P., Simperl, E., Gray, A., Sabou, M., Krötzsch, M., Lecue, F., Flöck, F., Gil, Y. (eds.) The Semantic Web – ISWC 2016, vol. 9981, pp. 200–217. Springer International Publishing, Cham (2016)

5. Euzenat, J., Le Duc, C.: Methodological guidelines for matching ontologies. In: Ontology engineering in a networked world, pp. 257–278. Springer (2012)
6. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer Berlin Heidelberg (2013)
7. Grüninger, M., Fox, M.S.: Methodology for the design and evaluation of ontologies. international joint conference on artificial inteligence. In: Workshop on Basic Ontological Issues in Knowledge Sharing. vol. 15, p. 34 (1995)
8. Hu, W., Chen, J., Zhang, H., Qu, Y.: Learning complex mappings between ontologies. In: Joint International Semantic Technology Conference. Springer (2011)
9. Jiang, S., Lowd, D., Kafle, S., Dou, D.: Ontology matching with knowledge rules. In: Transactions on Large-Scale Data-and Knowledge-Centered Systems (2016)
10. Lopez, V., Sabou, M., Motta, E.: Powermap: mapping the real semantic web on the fly. In: International Semantic Web Conference. pp. 414–427. Springer (2006)
11. Noy, N.F., Musen, M.A.: The prompt suite: interactive tools for ontology merging and mapping. International Journal of Human-Computer Studies 59(6) (2003)
12. Nunes, B.P., Mera, A., Casanova, M.A., Breitman, K.K., Leme, L.A.P.: Complex Matching of RDF Datatype Properties. In: 6th OM workshop (2011)
13. Parundekar, R., Knoblock, C.A., Ambite, J.L.: Linking and building ontologies of linked data. In: ISWC. pp. 598–614. Springer (2010)
14. Parundekar, R., Knoblock, C.A., Ambite, J.L.: Discovering concept coverings in ontologies of linked data sources. In: ISWC. pp. 427–443. Springer (2012)
15. Pinto, H.S., Martins, J.P.: Ontologies: How can They be Built? Knowledge and Information Systems 6(4), 441–464 (2004)
16. Qin, H., Dou, D., LePendu, P.: Discovering executable semantic mappings between ontologies. In: On the Move to Meaningful Internet Systems. pp. 832–849 (2007)
17. Ren, Y., Parvizi, A., Mellish, C., Pan, J.Z., van Deemter, K., Stevens, R.: Towards Competency Question-Driven Ontology Authoring. In: The Semantic Web: Trends and Challenges, vol. 8465, pp. 752–767 (2014)
18. Ritze, D., Meilicke, C., Šváb Zamazal, O., Stuckenschmidt, H.: A pattern-based ontology matching approach for detecting complex correspondences. In: 4th ISWC workshop on ontology matching. pp. 25–36 (2009)
19. Ritze, D., Völker, J., Meilicke, C., Šváb Zamazal, O.: Linguistic analysis for complex ontology matching. In: 5th workshop on ontology matching. pp. 1–12 (2010)
20. Rouces, J., de Melo, G., Hose, K.: Complex Schema Mapping and Linking Data: Beyond Binary Predicates. In: Proceedings of the WWW 2016 Workshop on Linked Data on the Web (LDOW 2016) (2016)
21. Thiéblin, E., Cheatham, M., Trojahn, C., Zamazal, O., Zhou, L.: The First Version of the OAEI Complex Alignment Benchmark. In: ISWC Posters and Demos (2018)
22. Thiéblin, E., Haemmerlé, O., Hernandez, N., Trojahn, C.: Task-Oriented Complex Ontology Alignment: Two Alignment Evaluation Sets. In: The Semantic Web. pp. 655–670. Lecture Notes in Computer Science, Springer (2018)
23. Thiéblin, E., Haemmerlé, O., Trojahn, C.: Complex matching based on competency questions for alignment: a first sketch. In: Ontology Matching Workshop (2018)
24. Unger, C., Forascu, C., Lopez, V., Ngomo, A.C.N., Cabrio, E., Cimiano, P., Walter, S.: Question answering over linked data (qald-4). In: Working Notes for CLEF 2014 Conference (2014)
25. Walshe, B., Brennan, R., O'Sullivan, D.: Bayes-recce: A bayesian model for detecting restriction class correspondences in linked open data knowledge bases. International Journal on Semantic Web and Information Systems 12(2), 25–52 (2016)
26. Zamazal, O., Svátek, V.: The Ten-Year OntoFarm and its Fertilization within the Onto-Sphere. Web Semantics: Science, Services and Agents on the World Wide Web 43, 46–53 (Mar 2017)