Reconciling Information in DBpedia through a Question Answering System

Elena Cabrio[1,2], Alessio Palmero Aprosio[3], and Serena Villata[1]

[1] INRIA Sophia Antipolis, France - firstname.lastname@inria.fr
[2] EURECOM, France
[3] Fondazione Bruno Kessler, Trento, Italy - aprosio@fbk.eu

Abstract. Results obtained querying language-specific DBpedia chapters SPARQL endpoints for the same query can be related by several heterogenous relations, or contain an inconsistent set of information about the same topic. To overcome this issue in question answering systems over language-specific DBpedia chapters, we propose the RADAR framework for information reconciliation. Starting from a categorization of the possible relations among the resulting instances, such framework: *(i)* classifies such relations, *(ii)* reconciles the obtained information using argumentation theory, *(iii)* ranks the alternative results depending on the confidence of the source in case of inconsistencies, and *(iv)* explains the reasons underlying the proposed ranking.

1 Introduction

In the Web of Data, it is possible to retrieve heterogeneous information items concerning a single real-world object coming from different data sources, e.g., the results of a single SPARQL query on different endpoints. These results may conflict with each other, or they may be linked by some other relation like a specification. The automated detection of the kind of relationship holding between different instances about a single object with the goal of reconciling them is an open problem for consuming in the Web of Data. In particular, this problem arises while querying the language-specific chapters of DBpedia, that may contain different information with respect to the English version. This issue becomes therefore particularly relevant in Question Answering (QA) systems exploiting DBpedia language-specific chapters as referential data set, since the user expects a unique (and possibly correct) answer to her factual natural language question.

In this demo, we propose the RADAR (ReconciliAtion of Dbpedia through ARgumentation) framework that: *i)* adopts a classification method to return the relation holding between two information items; *ii)* applies *abstract argumentation theory* [4] for reasoning about conflicting information and assessing the acceptability degree of the information items, depending on the kind of relation linking them; and *iii)* returns the graph of the results set, together with the acceptability degree of each information item, to motivate the resulting information ranking. We have integrated RADAR into the QA system QAKiS [1], that queries language-specific DBpedia chapters using a natural language interface.

2 RADAR: a Framework for Information Reconciliation

The RADAR framework (Fig. 1) takes as input a collection of results from the same SPARQL query raised against the language-specific DBpedia chapters SPARQL endpoints, and retrieves: *(i)* the sources proposing each particular element of the results set, and *(ii)* the elements of the results set themselves. The first module of RADAR (*Source confidence assignment score*, Fig. 1) takes each information source, and following two different heuristics, i.e., *Wikipedia page length* (the chapter of the longest language-specific Wikipedia page describing the queried entity is rewarded w.r.t. the others) and *entity geo-localization* (the chapter of the language spoken in the places linked to the page of the entity is rewarded with respect to the others), assigns a confidence degree to the source. Such metrics are summed, and normalized (0≤score≤1), where 0 is the less reliable chapter for a certain entity and 1 is the most reliable one. Such confidence degree will affect the reconciliation if inconsistencies arise: information proposed by the more reliable source will obtain a higher acceptability degree. The second module (*Relation classification module*, Fig. 1) starts from the results set, and it matches every element with all the other returned elements, detecting the kind of relation holding between this pair of elements, following the categorization of [3]. Such categories correspond to the linguistic phenomena (mainly discourse and lexical semantics relations) holding among heterogeneous values obtained querying two DBpedia language-specific chapters, given a certain subject and a certain ontological property. RADAR clusters the relations of *identity*, *disambiguated entity* and *coreference* into a unique category, called *surface variants* of the entity, and automatically detects such relation among two entities applying one of the following strategies: cross-lingual links (using WikiData), text identity (i.e., string matching), Wikipedia redirection and disambiguation pages. Moreover, RADAR integrates into a unique category *geo-specification* and *renaming*, and classifies a relation of this category when in GeoNames one entity results as contained in the other one. We also consider the alternative names gazette included in GeoNames, and geographical information extracted from English Wikipedia infoboxes, such as `Infobox former country`. Finally, RADAR clusters *meronymy*, *hyponymy*, *metonymy* and *identity:stage name* into a unique category, called *inclusion*, and detects it exploiting a set of features extracted from heterogeneous resources: MusicBrainz, NCF Thesaurus, DBpe-

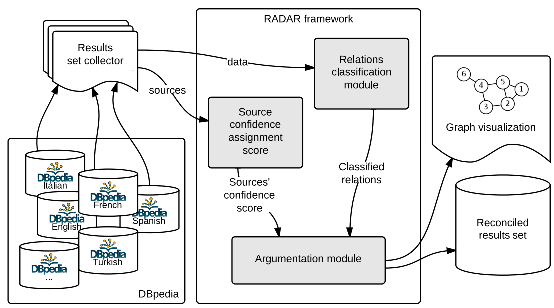

Fig. 1: RADAR framework architecture.

dia, WikiData and Wikipedia hierarchical information. Concerning inconsistent data in DBpedia language-specific chapters, RADAR labels a relation between entities/objects as negative, if every attempt to find one of the positive relations described above fails. The output consists in a graph composed by the elements of the results set connected with each other by the identified relations. Both the sources associated with a confidence score and the results set under the form of a graph are then provided to the third module of RADAR, the *Argumentation module* (Fig. 1). Its aim is to reconcile the results set: it considers all positive relations as a *support* relation and all negative relations as an *attack* relation, building a bipolar argumentation graph where each element of the results set is seen as an argument. Finally, adopting a bipolar fuzzy labeling algorithm [2] relying on the source's confidence to decide the acceptability of the information, the module returns the acceptability degree of each argument, i.e., element of the results set. RADAR provides as output: *i)* the acceptable elements (a threshold is adopted), and *ii)* the graph of the results set, i.e., the explanation about the choice of the acceptable elements returned.

Integrating RADAR into QAKiS. QAKiS addresses the task of QA over structured knowledge-bases (e.g., DBpedia) [1], where the relevant information is expressed also in unstructured forms (e.g., Wikipedia pages). It implements a relation-based match for question interpretation, to convert the user question into a query language (e.g., SPARQL), making use of relational patterns (automatically extracted from Wikipedia and collected in the WikiFramework repository) that capture different ways to express a certain relation in a given language. In QAKiS, the SPARQL query created after the question interpretation phase is sent to a set of language-specific DBpedia chapters SPARQL endpoints for answer retrieval. The set of retrieved answers from each endpoint is then sent to RADAR for answers reconciliation[1]. The user can select the DBpedia chapter she wants to query besides English (that must be selected as it is needed for Named Entity (NE) recognition), i.e., French or German. After writing a question or selecting it among the proposed examples, the user has to click on the tab *RADAR* where a graph with the answers provided by the different endpoints and the relations among them is shown. Each node has an associated confidence score, resulting from the fuzzy labeling algorithm. Moreover, each node is related to the others by a relation of support or attack, and a further specification of such relations according to the identified categories [3] is provided to the user as justification of the performed reconciliation and ranking.

To evaluate RADAR integration into QAKiS, we extract from QALD-2 data set[2] the questions currently addressed by QAKiS (i.e., questions containing a NE related to the answer through one single ontological property), corresponding to 58 questions (26 in the training, 32 in the test set). The discarded questions require either some forms of reasoning on data, aggregation from data sets other than DBpedia, involve n-relations, or are boolean questions. We submit

[1] A demo of RADAR integrated into QAKiS can be tested at `http://qakis.org`.
[2] `http://bit.ly/QALD2014`

such questions to QAKiS on the English, German and French DBpedia chapters. Since QALD-2 questions were created to query the English chapter only, it turned out that only in 25/58 cases at least two endpoints provide an answer (in all the other cases the answer is provided by the English chapter only, not useful for our purposes). For instance, given the question *Who developed Skype?* the English DBpedia provides *Skype Limited* as the answer, while the French one returns *Microsoft*. We evaluate the ability of RADAR to correctly classify the relations among the answers provided to the same query by the different language-specific endpoints, w.r.t. a manually annotated goldstandard (built according to [3]'s guidelines), carrying out two sets of experiments: *i)* we start from the answers provided by the different DBpedia endpoints to the 25 QALD questions, and we run RADAR on it; *ii)* we add QAKiS in the loop, meaning that the data we use as input for the argumentation module are directly provided by the system. We obtain the following results: RADAR achieves a precision/recall/f-measure of 1 in the classification of *surface form* and *inclusion* relations (overall positive: p/r/f=1); QAKiS+RADAR obtains p=1, r=0.60 and f=0.75 on *surface form*, p/r/f of 1 on *inclusion* (overall positive: p/r/f= 1/0.63/0.77). Since QALD-2 data was created to query the English chapter only, this small data set does not capture the variability of possibly inconsistent answers among DBpedia language-specific chapters. Only two categories of relations are present in this data, i.e., *surface forms*, and *inclusion*, and for this reason RADAR has outstanding performances when applied on the correct mapping between NL questions and SPARQL queries. When QAKiS is added into the loop, its mistakes in translating the NL question into the correct SPARQL query are propagated.

3 Future Perspectives

This demo improves the results of [2] as the categorization is more specific thus producing a more insightful explanation graph, and more performing techniques are applied to extract the relations. As future work, we will address a user evaluation to check whether QAKiS answer explanation suits data consumers' needs, and we will explore the possibility to leave the data consumer herself to assign the confidence degree to the sources depending on searched information.

References

1. Cabrio, E., Cojan, J., Gandon, F.: Mind the cultural gap: bridging language specific dbpedia chapters for question answering. In: Cimiano, P., Buitelaar, P. (eds.) Towards the Multilingual Semantic Web. Springer Verlag (2014)
2. Cabrio, E., Cojan, J., Villata, S., Gandon, F.: Argumentation-based inconsistencies detection for question-answering over dbpedia. In: NLP-DBPEDIA@ISWC (2013)
3. Cabrio, E., Villata, S., Gandon, F.: Classifying inconsistencies in dbpedia language specific chapters. In: LREC-2014 (2014)
4. Dung, P.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. 77(2), 321–358 (1995)