Artif Intell Rev (2013) 40:127–145
DOI 10.1007/s10462-012-9363-x

Building linked ontologies with high precision using subclass mapping discovery

Isabel F. Cruz · Matteo Palmonari · Federico Caimi · Cosmin Stroe

Published online: 9 November 2012

Abstract The creation of links between schemas of published datasets is a key part of the Linked Open Data (LOD) paradigm. The ability to discover these links "on the go" requires that ontology matching techniques achieve good precision and recall within acceptable execution times. In this paper, we add similarity-based and mediator-based ontology matching methods to the AgreementMaker ontology matching system, which aim to efficiently discover high precision subclass mappings between LOD ontologies. Similarity-based matching methods discover subclass mappings by extrapolating them from a set of high quality equivalence mappings and from the interpretation of compound concept names. Mediator-based matching methods discover subclass mappings by comparing polysemic lexical annotations of ontology concepts and by considering external web ontologies. Experiments show that when compared with a leading LOD approach, AgreementMaker achieves considerably higher precision and F-measure, at the cost of a slight decrease in recall.

Research partially supported by NSF Awards IIS-0812258, IIS-1143926, and IIS-1213013 and by the Intelligence Advanced Research Projects Activity (IARPA) via Air Force Research Laboratory (AFRL) contract number 2010-04569-00-02. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, AFRL, or the U.S. Government.

Matteo Palmonari: Work partially performed while visiting the ADVIS Laboratory at UIC.

I. F. Cruz (✉) · F. Caimi · C. Stroe
ADVIS Laboratory, Department of Computer Science, University of Illinois at Chicago,
Chicago, Illinois, USA
e-mail: ifc@cs.uic.edu

F. Caimi
e-mail: fcaimi@cs.uic.edu

C. Stroe
e-mail: cstroe1@cs.uic.edu

M. Palmonari
Department of Informatics, Systems and Communication, University of Milan-Bicocca, Milan, Italy
e-mail: palmonari@disco.unimib.it

Keywords Linked open data · Ontology matching · Data integration

1 Introduction

The linked data paradigm identifies a set of best practices to publish and share datasets on the web (Bizer et al. 2009). To integrate information, it is crucial to establish *correct* links among data, in what constitutes the *Linking Open Data (LOD)* cloud. The problem of establishing links between datasets (Volz et al. 2009; Bizer et al. 2009) is closely related to the problem of ontology matching that has been investigated in the semantic web and in the database communities (Euzenat and Shvaiko 2007; Rahm and Bernstein 2001).

The nature of the LOD cloud is changing due to the publication of semantic sensor data (Le-Phuoc et al. 2010), for example for traffic or environment monitoring (Valle et al. 2011; Gray et al. 2011). These are large and rapid evolving datasets, which will need efficient ontology matching strategies. In the LOD domain, tools like DBpedia Spotlight (Mendes et al. 2011) extract entities from unstructured documents at runtime. However, these tools consider only a single dataset and are not able to link the extracted data to other datasets and ontologies. Outside of the LOD domain, an "on the go" approach establishes a transitory agreement between parts of the agents' ontologies (Besana and Robertson 2005), for example, by modifying an existing ontology matching method, it can dynamically emphasize some mappings over others, so as to improve efficiency. A major difficulty in the creation of "on the go" strategies is that existing ontology matching systems do not yet meet the requirements of accuracy and of efficiency that are needed for the processing of large linked datasets. Therefore, the focus of this paper is on LOD ontology matching.

When performing ontology matching in the LOD setting, challenges include poor textual descriptions, a flat structure, cross-domain coverage, and concepts imported from external ontologies. Another challenge is that many ontology matching systems are better tailored to the discovery of equivalence relations. This is clearly a drawback in matching LOD ontologies because only a few equivalence relations can be found among concepts in highly heterogeneous ontologies. Therefore, the capability to discover other relations, such as subclass relations, is crucial.

Prior work in LOD ontology matching has been performed by the BLOOMS system (Jain et al. 2010). This work has introduced a new matching approach based on searching Wikipedia pages related to the ontology terms: the categories extracted from these pages are then organized into graphs and used to match the ontology terms. In the LOD setting, BLOOMS performs better than systems that have been designed to work in "classic" ontology matching settings based on equivalence mappings, such as those in the Ontology Alignment Evaluation Initiative (OAEI) (Euzenat et al. 2009, 2010, 2011). In contrast, those systems outperform BLOOMS in the "classic" setting (Euzenat et al. 2011). Therefore, none of these systems is a top performer in both "classic" and LOD settings.

In this paper, we extend the AgreementMaker ontology matching system,[1] (Cruz et al. 2009a,b), which has obtained some of the best results in the OAEI (Cruz et al. 2009c, 2010, 2011b), with the objective of testing its viability in the LOD domain. In this paper we address the following two questions: (1) How can a top performing system like AgreementMaker be extended to handle mappings other than equivalence mappings? (2) Can AgreementMaker achieve top accuracy and efficiency in the LOD domain?

[1] http://www.agreementmaker.org.

To address the first question, we develop four new ontology matching methods. A first category of matching methods compares directly a source and a target ontologies and includes: (i) the *Equivalence Mappings Extension* method, which uses a set of equivalence mappings discovered with high confidence to infer subclass and superclass mappings, and (ii) the *Compound Noun Analysis* method, which discovers subclass and superclass mappings by analyzing compound names that are often used to identify ontology concepts. A second category of matching methods exploit third party ontologies that are used as mediators and includes: (i) the *Distance-based Polysemic Lexical Comparison* method, which automatically annotates ontology concepts with lexical concepts taken from a background terminology and then compares these lexical annotations to discover subclass and superclass mappings, and (ii) the *Global Matching* method that infers subclass and superclass mappings by looking at how the concepts are used in popular web ontologies.

As for the second question, we show that our approach achieves results in LOD ontology matching that are considerably better than other ontology matching systems in terms of precision and F-measure. In terms of recall, our approach is the second best, just slightly below that of the BLOOMS system. In addition, our approach is more efficient in terms of execution time than BLOOMS and has the advantage that it consists of methods that can be integrated with an existing ontology matching system. To the best of our knowledge, AgreementMaker is currently the only system that achieves top performance both in the "classic" and LOD domains.

In comparison with the preliminary version of this paper (Cruz et al. 2011a), we have introduced the following modifications:

- we have four specialized matching methods, which replace the two previous methods
- we improve the capability to discover subclass mappings by analyzing compound nouns, using a stand-alone method
- we extend the use of a background terminology by exploiting polysemic lexical annotations and the distance between concepts in the terminology hierarchy
- we extend the experimental evaluation providing an in-depth analysis of the results achieved by each method

The paper is organized as follows. The proposed methods to improve ontology matching in the LOD domain are described in Sects. 2, 3, and 4. The experimental evaluation of the proposed approach, based on previously proposed reference alignments (Jain et al. 2010) is discussed in Sect. 5. Related work is discussed in Sect. 6. Finally, concluding remarks end the paper in Sect. 7.

2 Matching LOD ontologies: approach overview

Given a *source ontology* S and a *target ontology* T, a *mapping* is a triple $\langle c_S, c_T, r\rangle$ where c_S and c_T are concepts in S and T, respectively, and r is a semantic relation that holds between c_S and c_T.

A set of mappings is called an *alignment*. A *reference alignment* or *gold standard* is an alignment found by experts, against which the accuracy of other alignments, as measured in terms of precision and recall, can be determined. In *ontology matching* one attempts to find as many accurate mappings as possible using *matching algorithms*, which we call *matchers*.

We consider three types of semantic relations: *subclass of* ($\sqsubseteq$), *superclass of* ($\sqsupseteq$), and *equivalence* ($\equiv$), all interpreted according to their usual semantics in OWL (Staab and

Table 1 Categorization of our matching algorithms

Matcher	Acr	Cat	Rel	Lex
Equivalence mappings extension	EME	dir	$\sqsubseteq, \sqsupseteq, \equiv$	
Compound noun analysis	CNA	dir	$\sqsubseteq, \sqsupseteq$	✓
Distance-based polysemic lexical comparison	DPLC	med	$\sqsubseteq, \sqsupseteq$	✓
Global matching	GM	med	$\sqsubseteq, \sqsupseteq$	

Studer 2004). Given these relations, we can define three types of mappings: $\langle c_S, c_T, \sqsubseteq\rangle$, meaning that c_S is a *subclass* of c_T, $\langle c_S, c_T, \sqsupseteq\rangle$ meaning that c_S is a *superclass* of c_T, and $\langle c_S, c_T, \equiv\rangle$, if, and only if, $\langle c_S, c_T, \sqsubseteq\rangle$ and $\langle c_S, c_T, \sqsupseteq\rangle$. In this case, c_S and c_T are *equivalent* classes.

Our approach to matching LOD ontologies integrates four methods grouped into two main categories. Each method has been implemented in a matcher and addresses a particular matching pattern as explained below.

Direct mapping discovery. These methods discover mappings between the source and target ontologies by directly comparing their concepts using a *similarity metric*. The first method, called *Equivalence Mappings Extension (EME)*, uses a *similarity metric* to discover a set of equivalence mappings, from which two sets of subclass and superclass mappings are inferred. The second method, called *Compound Noun Analysis (CNA)*, discovers subclass and superclass mappings by analyzing the compound names that are used as local names in several concepts (e.g., *SportsEvent* is mapped to *Event* by a subclass relation).

Mediator-based mapping discovery. These methods make use of third-party ontologies playing the role of mediators to discover subclass and superclass mappings between the source and target ontologies. The first method, called *Distance-based Polysemic Lexical Comparison (DPLC)*, is based on the lexical annotation of ontology concepts with terminology concepts organized in a hierarchy. Subclass and superclass mappings between ontology concepts are discovered by comparing their lexical annotations. This method adopts an approximate matching technique that handles polysemic annotations of ontology concepts, which associate more than one terminology concept to each ontology concept. The distance between the terminology concepts in the hierarchy is also considered. The second method, called *Global Matching (GM)*, discovers subclass and superclass mappings that involve concepts defined in external ontologies by looking at how these concepts are used in other popular web ontologies.

The four proposed matchers and their main features are summarized in Table 1. In the *acronym (Acr)* column, the acronyms used in the paper for each matcher are listed. In the *category (Cat)* column, we report if a matcher follows a *direct (dir)* or *mediator-based (med)* matching approach. In the *relations (Rel)* column, we report the kind of semantic relations considered. Finally, in the *lexical (Lex)* column, we report which matchers use lexical analysis as their main component.

The alignment $Alignment(S, T)$ between a source ontology S and a target ontology T is defined as the union of the sets of mappings determined by the four matchers, *EME, CNA, DPLC,* and *GM*.

In the rest of the paper, we will use the following notation. Given a matcher M, M^r denotes the mappings discovered by M whose relation is r (as an example, $EME^{\sqsubseteq}$ denotes the set of *subclass* mappings discovered by EME).

3 Similarity-based mapping discovery

Equivalence mappings are discovered by evaluating a similarity value in the interval [0,1] between every pair $\langle c_S, c_T \rangle$ of source and target concepts, denoted $sim(c_S, c_T)$. The similarity value is a measure of the confidence with which we believe that the two concepts are semantically equivalent. We use the *Advanced Similarity Matcher (ASM)* to compute the similarity $sim(c_S, c_T)$ between two concepts c_S and c_T. ASM is a very efficient matcher that evaluates the string-based similarity between two concepts using their local names and their labels (Cruz et al. 2010). Two concepts are considered equivalent when their similarity is higher than a threshold $th^{\equiv}$.

We slightly modified ASM to detect different spellings of the same word, for example $(Organization, Organisation)$ and $(Theater, Theatre)$. These apparently small differences are not always captured by string similarity algorithms, but simple linguistic rules like this one significantly improve the capability to discover equivalence mappings.

3.1 Equivalence Mappings Extension

The Equivalence Mappings Extension (EME) matcher computes the similarity values between all the possible pairs of concepts and stores the results in a similarity matrix. For each pair of concepts and a threshold $th^{\equiv}$, such that $sim(c_S, c_T) \geq th^{\equiv}$, the mapping $\langle c_S, c_T, \equiv \rangle$ is included in the set of equivalence mappings $EME^{\equiv}$.

Starting with $EME^{\equiv}$, we build $EME^{\sqsubseteq}$ and $EME^{\sqsupseteq}$ by considering subclasses and superclasses of the concepts c_S and c_T that appear in the mappings $\langle c_S, c_T, \equiv \rangle \in EME^{\equiv}$. We add to the set $EME^{\sqsubseteq}$ (respectively, $EME^{\sqsupseteq}$) all the triples $\langle x_S, c_T, \sqsubseteq \rangle$ (respectively, $\langle c_S, x_T, \sqsupseteq \rangle$) such that x_S is a subclass of c_S (respectively, c_T is a subclass of x_T).

The selection of the equivalence mappings must be even more accurate in the LOD domain than what is required in traditional ontology matching scenarios (Euzenat et al. 2011); this is a consequence of the importance of subclass and superclass mappings. When equivalence mappings are used to infer subclass mappings, a wrongly determined equivalence mapping can propagate an error to all the derived mappings. For this reason, in the LOD domain we set a very high threshold, e.g., 0.95, while in several other domains thresholds in the range [0.6, 0.8] are usually adopted (Cruz et al. 2010).

3.2 Compound Noun Analysis

When the names of the concepts in the ontologies are *compound*, that is, when they are formed by multiple words, matchers such as ASM, which is highly specialized for the equivalence relation, are not able to capture other relations that are implicitly specified in the compound. For example, *SportsEvent* denotes a narrower concept than *Event*, thus a subclass relation should be directly inferred from their names (under the assumption that the two concepts share the same meaning for the term *Event*).

An exhaustive classification of compounds in English has been proposed and is shown in Fig. 1 (Plag 2003). The majority of the compounds has a *modifier-head* structure, where the *head*, the most important unit, usually determines the gender, part-of-speech, and the general meaning. This general meaning is then modified by the other terms, restricting the meaning of the compound to a more specific concept. In the previous example, *Event* is the head and *Sports* is a modifier.

When the names of the concepts to be matched are compound, we use a best effort approach that produces good results in practice. We consider only *endocentric* compounds,

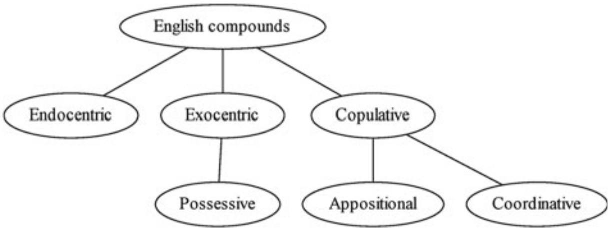

Fig. 1 Classification of compounds (i.e., compound words) in English (Plag 2003)

which are compounds with the head occurring in the compound itself (e.g., *Sports Event*), because they are the vast majority in English and cover up to 78% of the compounds used in schema and ontology concept names according to a recent study (Sorrentino et al. 2010). For these compounds, we are interested in detecting the head, as it provides meaningful information for inferring subclass relations. It has been observed that, in English, the heads of endocentric compounds always occur on their right-hand side (Williams 1981). We use this knowledge to extract the heads and then attempt to find correspondences between these main nouns and the names of the concepts using ASM. Based on these correspondences, we extrapolate subclass and superclass mappings. In particular, let $head(c)$ be the head of a compound denoting the concept c. If $sim(head(c_S), c_T)) \geq th^{\equiv}$, then $\langle c_S, c_T, \sqsubseteq\rangle \in CNA^{\sqsubseteq}$; if $sim(c_S, head(c_T)) \geq th^{\equiv}$, then $\langle c_S, c_T, \sqsupseteq\rangle \in CNA^{\sqsupseteq}$.

4 Mediator-based mapping discovery

In this paper, we consider two different types of mediators, namely background terminologies and web ontologies. *Web ontologies* are ontologies represented in a semantic web language (e.g., RDFS or OWL (Staab and Studer 2004)) and available on the web.

A *background terminology* is any knowledge structure organized in a concept hierarchy. It can be represented by a triple $O^T = (C, T, \preceq)$, where C is a set of *terminology concepts*, T is a set of *terms* (also called *labels*) and $\preceq$ is a *hyponymy* relation defined by a partial order over C. Given two terminology concepts w_1 and w_2, the relation $w_1 \preceq w_2$ means that w_1 is *more specific* than w_2. In this case, w_1 is a *hyponym* of w_2, and, conversely, w_2 is a *hypernym* of w_1. Each concept is associated with a set of synonymous terms (*synonyms*). Conversely, a term can be associated with one or more concepts (*polysemy*).

Background terminologies encompass knowledge structures such as lexicons and other taxonomies where multiple labels are associated with a concept. In this paper, we use WordNet as background terminology, whose concepts are called *synsets*, each one usually associated with more than one term (synonym).

Although background terminologies and web ontologies share a similar hierarchical structure, the semantics of the relations on which their respective hierarchies are based is different: while in a web ontology $c_1 \sqsubseteq c_2$ means that c_1 is subclass of c_2, that is, every instance of c_1 is also an instance of c_2, in a terminology the hyponym relation cannot be assumed to have such formal semantics. In other words, it can be the case that $w_1 \preceq w_2$ while $w_1 \not\sqsubseteq w_2$. For example, the terminology concept *Hazard* is defined in WordNet as "a source of danger; a possibility of incurring loss of misfortune". Following the hypernym hierarchy, *Hazard* has *Physical Entity* among its hypernyms. However, "drinking alcohol" (mentioned in WordNet as an example of *Hazard*) can hardly be considered an instance of *Physical Entity*. Another

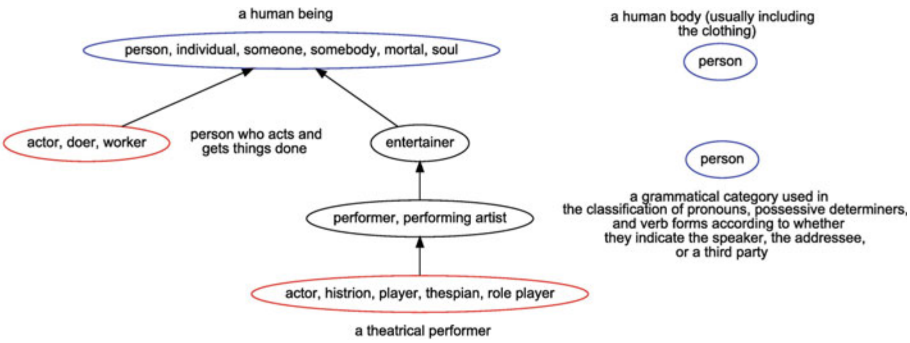

Fig. 2 WordNet synsets for the ontology concepts *Actor* (source) and *Person* (target). Each ellipse represents a WordNet synset with its set of terms. The synsets associated with the source and target concepts are highlighted respectively in *red* and *blue*. The *arrows* represent the hyponym relation. (Color figure online)

difference between background terminologies and web ontologies is the lack of a systematic coverage of polysemy and synonymy in web ontologies. The consideration of these important differences leads to the design of different matching methods depending on the type of mediator, as further described in the rest of this section.

4.1 Distance-based Polysemic Lexical Comparison

We compare every concept of the source ontology with every concept in the target ontology. The key idea of our algorithm is that given a source concept c_S lexically annotated with a terminology concept w^{c_S} and a target concept c_T lexically annotated with w^{c_T}, we can add a subclass mapping $\langle c_S, c_T, \sqsubseteq\rangle$ (respectively, a superclass mapping $\langle c_S, c_T, \sqsupseteq\rangle$) when $w^{c_S} \preceq w^{c_T}$ (respectively, $w^{c_T} \preceq w^{c_S}$) holds in the terminology.

However the simple idea sketched above encounters two problems:

1. It can be very difficult to annotate an ontology concept with exactly one terminology concept for two reasons. The first one is that information needed to automatically disambiguate among several candidate annotations can be inadequate. For example, in Fig. 2 there are three sets of synonyms associated with the concept *Person* (highlighted in blue), and there is no empirical evidence of one being more appropriate than the others. Therefore, they should all be considered in the following steps. The second reason is that a terminology can provide several concepts with similar meaning, which can all be considered correct annotations for the ontology concept (Po and Sorrentino 2011). In Fig. 2, the two sets of synonyms associated with the concept *Actor* (highlighted in red) are very similar and can be both considered correct annotations for the ontology concept. Therefore, the matching algorithm has to handle the case in which concepts are associated with multiple lexical annotations.
2. In general, the semantics of the relation $\preceq$ is different from the semantics of the subclass relation $\sqsubseteq$. Therefore, the more distant two terminology concepts are in the terminology hierarchy, the higher the probability that they are weakly related, and, therefore, the higher the probability that the inferred mapping among the ontology concepts is wrong. The *distance* (length of the path) between two lexical annotations on the terminology hierarchy can be used to assign a confidence score to the inferred mapping.

We address these two problems with an algorithm consisting of three steps.

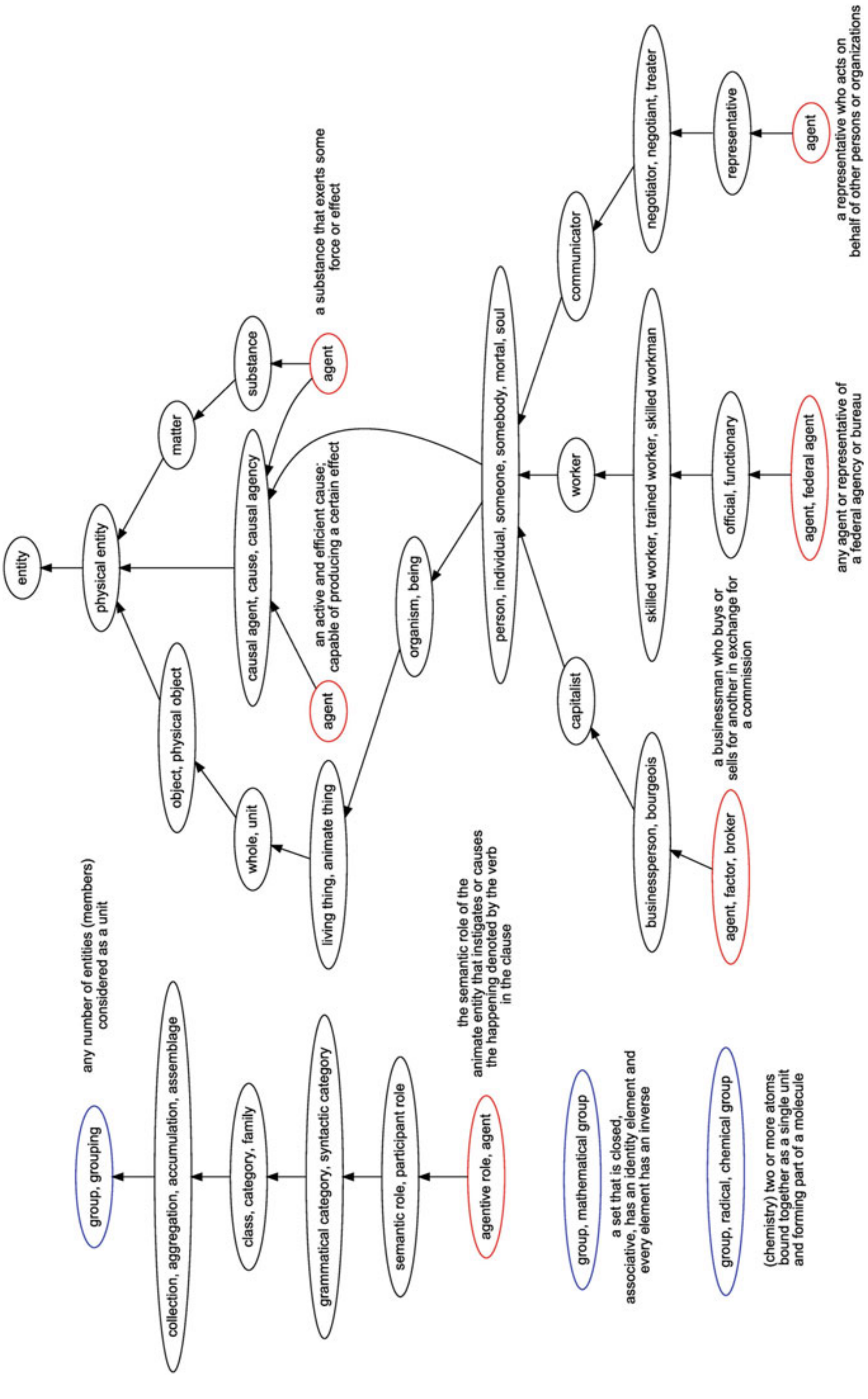

Fig. 3 WordNet synsets for *Agent* and *Group*

Step 1. Polysemic Lexical Annotation with Word Sense Disambiguation. Each concept in the source (respectively, target) gets associated with a set of concepts in the background terminology. This association is made through the concept labels: every time a label matches exactly a concept in the source (respectively, target) ontology, then that terminology concept becomes associated with the source (respectively, target) concept. Given a concept c, the set of the terminology concepts associated with it is denoted by BST_c (for *Background Synonym Terminology*). In Figs. 2 and 3, two graphs involving the terminology concepts are shown, where the elements of BST_{c_S} (*Actor* and *Agent*) are highlighted in red and the BST_{c_T} concepts (*Person* and *Group*), are highlighted in blue.

However, to improve the accuracy of lexical annotation, we apply *word sense disambiguation* techniques (Po and Sorrentino 2011). Some concepts in the ontologies have a textual description (usually included in *rdfs:comment*), while in WordNet all the sets of synonyms are described in a *definition*. We create a virtual document associated with each concept and with each synset, after performing stop word removal and stemming. The virtual documents are then compared using a vector space model approach based on the cosine similarity measure. These techniques were already implemented in one of our matchers called Vector-based Multi-word Matcher (VMM), extensively used in the OAEI (Cruz et al. 2009c). In addition to comments and definitions, we also included in the documents the first level of the concepts' superclasses, since they proved to be particularly useful for disambiguation. After the similarity values are computed, the actual disambiguation is performed. If the degree of similarity between a concept and a related synset is higher than a given threshold, only those will be kept for further processing, thus narrowing the set BST_c into a subset $\overline{BST}_c$. The threshold has been experimentally set to 0.3, a high value for cosine similarity. This leads to an improvement in precision, while not penalizing recall.

Step 2. Background Hypernym Terminology Construction. Each concept in the source (respectively, target) gets associated with a set of hypernyms from the background terminology. This association is made by means of the previously built sets of synonyms. Given a concept c, we consider each concept in $\overline{BST}_c$ and extract its hypernyms in the background terminology. Finally, we take the union of all such sets, thus obtaining a set for each concept c denoted BHT_c (for *Background Hypernym Terminology*).

Step 3. Mapping Inference. We use the sets obtained in the previous two steps to build the sets of subclass and superclass mappings denoted respectively by $DPLC^{\sqsubseteq}$ and $DPLC^{\sqsupseteq}$.

Our mediator-based approach relies on the conversion of hypernym relations into subclass relations, the latter being interpreted according to their well-known OWL semantics. We denote the correct (e.g., as determined by a pool of experts) annotations for the source concept c_S and of the target concept c_T as w^{c_S} and w^{c_T}, respectively. We then define the *hyponym-to-subclass conversion factor* (*hsc*) as the probability that a source concept c_S is a subclass of a target concept c_T, given that w^{c_S} is a *direct hyponym* of w^{c_T}:

$$hsc = P(c_S \sqsubseteq c_T | w^{c_S} \preceq^1 w^{c_T}) \quad (1)$$

where $\preceq^1$ denotes the direct hyponymy relation. We note that the *hsc* factor can change depending on the terminology. We empirically estimated *hsc* to be 0.9 in WordNet, based on the manual inspection of a few dozen branches of the WordNet concept hierarchy.

Now we can define the metric that computes the confidence value associated with the existence of a subclass mapping between a source and a target concept. We call this confidence value the *single-annotation subclass evidence score*, denoted by $saScore(c_S^{w^{c_S}}, c_T^{w^{c_T}})$. This metric, which is based on the propagation of the *hsc* factor along the path between w^{c_S} and w^{c_T} with length denoted by $dist(w^{c_S}, w^{c_T})$, is computed as follows:

$$saScore(c_S^{w^{c_S}}, c_T^{w^{c_T}}) = \begin{cases} P(c_S \sqsubseteq c_T | w^{c_S} \preceq w^{c_T}) & \text{if } w^{c_S} \preceq w^{c_T} \\ 0 & \text{if } w^{c_S} \npreceq w^{c_T} \end{cases} \quad (2)$$

where $P(c_S \sqsubseteq c_T | w^{c_S} \preceq w^{c_T})$ is defined as follows:

$$\begin{aligned} P(c_S \sqsubseteq c_T | w^{c_S} \preceq w^{c_T}) &= \prod_{i=1}^{dist(w^{c_S}, w^{c_T})-1} P(c_S \sqsubseteq c_T | w_i \preceq^1 w_{i+1}) \quad (3) \\ &\quad \text{where } w_1 = w^{c_S}, w_{dist(w^{c_S}, w^{c_T})} = w^{c_T} \\ &= hsc^{dist(w^{c_S}, w^{c_T})} \end{aligned}$$

We also compute $saScore(c_T^{w^{c_T}}, c_S^{w^{c_S}})$ by adapting Eqs. 2 and 3 accordingly.

Finally, we need to consider that according to the lexical annotation strategy adopted, an ontology concept may be annotated with more than one terminology concept. We therefore define a *polysemic subclass evidence score* that aggregates the single-annotation subclass evidence scores for all the lexical annotations. The polysemic subclass evidence score $polyScore(c_S, c_T)$ is defined as follows:

$$polyScore(c_S, c_T) = \frac{\sum_{w_i \in \overline{BST}_{c_S}, w_j \in BHT_{c_T}} saScore(c_S^{w_i}, c_T^{w_j})}{log(|BHT_{c_T}|)} \quad (4)$$

We use a normalization factor in the denominator that is based on the size of the Background Hypernym Terminology, BHT_{c_T}, which is associated with the target concepts. In fact, the bigger this set is, the higher the probability of finding matchings between sets of synonyms and hypernyms. Because the size of these sets grows rapidly when the length of the paths increases, we use the logarithm of BHT_{c_T}. We also compute $polyScore(c_T, c_S)$ by adapting Eq. 4 accordingly.

In Fig. 2, there are two paths connecting the source and target terminology concepts, respectively associated with the source concept *Actor* and with the target concept *Person*. The first path (of length one) gets associated with an *saScore* of 0.9, while the second path (of length three) gets associated with a value of 0.729. These values are then added and normalized by applying the natural logarithm of BHT_{c_S}, which in this case is 10 (the hypernyms of *Person* are not shown for simplicity). The overall score (0.707) is above the threshold we experimentally set, and therefore a mapping between the ontology concepts *Actor* and *Person* will be included in $DPLC^{\sqsubseteq}$.

In Fig. 3, there is only one path connecting the source and target terminology concepts, even though the graph is significantly larger than in the previous example. This path (of length five) gets associated with an *saScore* of 0.59. After normalization, the overall score obtained (0.186) is below the threshold, and therefore that mapping will not be included in $DPLC^{\sqsubseteq}$.

The polysemic subclass evidence score can be adopted to infer both subclass and superclass relations. In fact, given a subclass score threshold $th^{\sqsubseteq}$, the set of subclass mappings and superclass mappings returned by this matcher can be defined as follows:

$$DPLC^{\sqsubseteq} = \{\langle c_S, c_T, \sqsubseteq\rangle | polyScore(c_S, c_T) \geq th^{\sqsubseteq} \text{ and } polyScore(c_S, c_T) \geq polyScore(c_T, c_S)\}$$
$$DPLC^{\sqsupseteq} = \{\langle c_S, c_T, \sqsupseteq\rangle | polyScore(c_T, c_S) \geq th^{\sqsubseteq} \text{ and } polyScore(c_T, c_S) \geq polyScore(c_S, c_T)\} \quad (5)$$

4.2 Global Matching

LOD ontologies often use several concepts (e.g., *foaf:Person* in the ontology of the Semantic Web Conference) that are imported from other ontologies and therefore need to be considered in the matching process. In traditional ontology matching scenarios, this kind of interlinking is rarer.

The *Global Matching (GM)* method is introduced to improve matching over external concepts. This method is based on the fact that several external concepts used in LOD ontologies, such as *wgs84_pos:SpatialThing* in the GeoNames ontology, are used across different ontologies. Such external concepts can help in discovering additional mappings. For example, it is possible to determine that a mapping exists between *dbpedia:Person* and *wgs84_pos:SpatialThing* if *foaf:Person* has been defined as a subclass of *wgs84_pos:SpatialThing* elsewhere.

Our GM method works as follows. For each source concept c_S that has been imported from an external ontology E, we search across several LOD ontologies for all concepts that have been defined as subclasses of c_S and we match these concepts with the concepts of the target ontology using the ASM matcher. We proceed similarly for each target concept c_T. More specifically, if there is in some external ontology E a concept x_E such that x_E has been defined as subclass of c_S (respectively, c_T) and for some concept c_T (respectively, c_S) $sim(x_E, c_T) \geq th^{\equiv}$ (respectively, $sim(c_S, x_E) \geq th^{\equiv}$), then $\langle c_S, c_T, \sqsupseteq\rangle \in GM^{\sqsupseteq}$ (respectively, $\langle c_S, c_T, \sqsubseteq\rangle \in GM^{\sqsubseteq}$).

The external ontologies that we use to search for external concepts are listed in a registry. We included in the registry web ontologies that have either been defined by a recognized institution such as the W3C consortium (e.g., Event Ontology,[2] WGS84 Geo Positioning,[3] and Media Ontology)[4] or are well known and used by a wide community of users (e.g., DBPedia,[5] FOAF,[6] and Freebase).[7]

These ontologies often reuse the most important concepts of third party ontologies. For this reason, they provide good background knowledge, because of the shared external concepts.

5 Experimental results

Table 2 lists the ontologies that we have used for our experiments, which are the same that were considered by the BLOOMS system[8] (Jain et al. 2010). We note that no other benchmark has been has been set for the LOD domain. The table shows the number of concepts in the ontologies and the number of external ontologies that they import. The evaluation settings

2 http://motools.sourceforge.net/event/event.html.

3 http://www.w3.org/2003/01/geo/wgs84_pos.

4 http://www.w3.org/TR/mediaont-10/.

5 http://dbpedia.org/ontology/.

6 http://xmlns.com/foaf/spec/.

7 http://rdf.freebase.com/rdf/base.fbontology.

8 http://wiki.knoesis.org/index.php/BLOOMS.

Table 2 Ontologies in the experimental dataset (Jain et al. 2010)

Ontology	Id	# Classes	# Imported ontologies
AKT portal	AKT	169	1
BBC program	BBC	100	2
DBpedia	DBp	257	0
FOAF	FOAF	16	0
GeoNames	GN	10	0
Music ontology	MO	123	8
Semantic web conference	SWC	172	0
SIOC	SIOC	15	0

Table 3 Comparison of S-Match, Aroma, BLOOMS (Jain et al. 2010), and AgreementMaker, with maximum values for each matching task in bold

Matching task	S-Match			AROMA			BLOOMS			AgreementMaker		
	Prec	Rec	F-m	Prec	Rec	F-m	Prec	Rec	F-m	Prec	Rec	F-m
FOAF-DBp	0.11	0.40	0.17	0.33	0.04	0.07	0.67	0.73	0.70	**0.80**	**0.90**	**0.85**
GN-DBp	0.23	**1.00**	0.37	0.00	0.00	0.00	0.00	0.00	0.00	**0.32**	0.73	**0.44**
MO-BBC	0.04	0.28	0.07	0.00	0.00	0.00	**0.63**	**0.78**	**0.70**	0.56	0.27	0.36
MO-DBp	0.08	0.30	0.13	0.45	0.01	0.02	0.39	**0.62**	0.48	**0.87**	0.46	**0.60**
SWC-AKT	0.06	0.40	0.10	0.38	0.03	0.06	0.42	**0.59**	**0.49**	**0.52**	0.41	0.46
SWC-DBp	0.15	0.50	0.23	0.27	0.01	0.02	0.70	**0.40**	**0.51**	**0.71**	0.39	0.50
SIOC-FOAF	0.52	0.11	0.18	0.30	0.20	0.24	0.55	**0.64**	**0.59**	**0.71**	0.45	0.55
Average	0.17	0.43	0.24	0.25	0.04	0.07	0.48	**0.54**	0.51	**0.64**	0.52	**0.57**

consist of seven matching tasks, involving different types of comparisons. For example, the Music Ontology and the BBC Program ontology are both related to entertainment, whereas some other comparisons involve general purpose ontologies, such as DBpedia.

In this section, we first compare the results obtained by our system with the results obtained by other systems for the seven matching tasks. We then provide an in-depth analysis of each matcher that we used in our system. Finally, we provide a discussion of our results, which we believe is of interest for future research in this domain.

Comparison with other systems. Table 3 shows the comparison between the results obtained by AgreementMaker and the results previously obtained for the S-Match, AROMA, and BLOOMS ontology matching systems (Jain et al. 2010). We are omitting the baseline results (Alignment API) and the results that were reported for other systems (OMViaUO, and RiMOM) because their results are not competitive in the LOD domain (Jain et al. 2010).

As can be seen in Table 3, our system achieves the best average precision and F-measure by a large margin (respectively, 16 and 6 %). In terms of recall, BLOOMS is number one followed closely by AgreementMaker. We comment next on the results obtained for each task.

Task 1. For the FOAF-DBpedia matching task, our system has the best values for precision and recall. In particular, non-trivial mappings are discovered by our global matching

Table 4 Comparison of AgreementMaker with its previous version (Cruz et al. 2011a), with maximum values for each matching task in bold

Matching task	AgreementMaker (2010)			AgreementMaker		
	Prec	Rec	F-m	Prec	Rec	F-m
FOAF-DBp	0.72	0.80	0.76	**0.80**	**0.90**	**0.85**
GN-DBp	0.26	0.68	0.38	**0.32**	**0.73**	**0.44**
MO-BBC	0.48	0.16	0.24	**0.56**	**0.27**	**0.36**
MO-DBp	0.62	0.40	0.49	**0.87**	**0.46**	**0.60**
SWC-AKT	0.48	**0.43**	0.45	**0.52**	0.41	**0.46**
SWC-DBp	0.58	0.35	0.44	**0.71**	**0.39**	**0.50**
SIOC-FOAF	0.56	0.41	0.47	**0.71**	**0.45**	**0.55**
Average	0.53	0.46	0.49	**0.64**	**0.52**	**0.57**

technique described in Sect. 3, which allows us to find mappings using external ontologies and to propagate them through the subclasses of the involved concepts.

Task 2. For the GeoNames-DBpedia matching task, BLOOMS is not able to find mappings. This is because the GeoNames ontology has very little information that is contained in the ontology proper, as the actual categories are encoded in properties at the instance level. However, S-Match has a perfect recall (100 %), though precision is low (23 %). The use of our global matching technique is the main reason why AgreementMaker outperforms all the other systems.

Task 3. For the Music Ontology-BBC program task, BLOOMS obtains the best results, with AgreementMaker second. BLOOMS uses Wikipedia while we use WordNet, a generic background ontology. Wikipedia is very well suited for this kind of ontologies, because it covers the specific vocabulary of the ontologies being matched.

Task 4. For the Music Ontology-DBpedia matching task, and in contrast with the previous task, our results are better than those of BLOOMS in terms of F-measure. While BLOOMS achieves slightly higher recall, the precision achieved by AgreementMaker is significantly higher. Our system presents only mappings for which it is very confident, thus favoring precision, while BLOOMS clearly favors recall. The next best system, S-Match, obtains reasonable recall (30 %), albeit at the cost of very low precision (8 %).

Task 5. For the Semantic Web Conference-AKT Portal matching task in the scientific publications domain, we notice again that BLOOMS favors recall while AgreementMaker favors precision. S-Match again favors recall at the cost of very low precision, while Aroma favors precision at the cost of very low recall.

Task 6. For the Semantic Web Conference-DBpedia matching task, BLOOMS and AgreementMaker achieve very similar good results. The conference domain is the same used in the OAEI, on which both systems perform well. S-Match has an interesting recall (50%), but low precision (15%).

Task 7. For the SIOC-FOAF matching task, both general linguistic understanding and specific domain vocabulary are needed, because SIOC is an ontology related to online communities.

Table 5 Execution times (in seconds) of the matching process (loading, similarity-based, mediator-based, and total)

Matching task	Load	SB	MB	Total
FOAF-DBpedia	6.9	3.1	1.7	11.7
GeoNames-DBpedia	6.6	1.5	1.6	9.8
Music ontology-BBC program	16.0	3.7	4.7	24.4
Music ontology-DBpedia	26.3	18.2	7.5	52.1
Semantic web conference-AKT portal	3.5	2.1	2.8	8.3
Semantic web conference-DBpedia	7.9	8.1	2.4	18.5
SIOC–FOAF	0.1	0.2	1.7	2.0

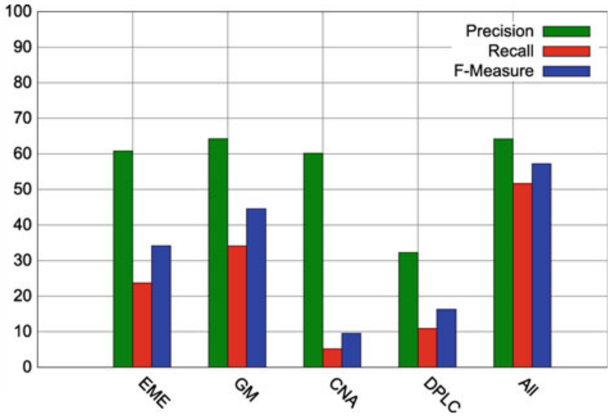

Fig. 4 Analysis of the effectiveness of each matcher

AgreementMaker leads in precision followed by BLOOMS and S-Match (respectively, 71, 55, and 52 %), while BLOOMS significantly leads in recall because it is based on Wikipedia.

Table 5 shows the total execution times of the AgreementMaker matching process in the seven tasks as well as the times for the different subtasks, namely loading and mapping discovery using the similarity-based (SB) and mediator-based (MB) methods. We note that the total execution time never exceeds one minute, even when the largest ontologies such as the Music Ontology and DBpedia are being matched (with 123 and 257 classes, respectively).

We compared the performances of the Semantic Web Conference-AKT Portal matching task between AgreementMakerand BLOOMS, the second best system. While BLOOMS took 2 hours and 3 minutes, performed the same task in only 8.3 seconds. We ran our experiments using an Intel Core2 Duo T7500 2.20GHz with 2GB RAM and Linux kernel 2.6.32-30 32 bits. We note that BLOOMS uses a web service to access the Wikipedia pages and that the gap between AgreementMakerand BLOOMS was even wider for the other tasks.

Analysis of the results. Figure 4 shows the results (precision, recall, and F-measure) achieved by our system overall (All) and by each of the methods (GM, EME, DPLC, CNA) so that we can analyze the contributions of each of them. *Global Matching* (GM) has the best recall. This is due to the fact that external concepts usually have a high number of subclasses and that in some of the seven matching tasks, most of the mappings involve external concepts. In addition, GM it is the best method in terms of precision.

The *Equivalence Mappings Extension* (EME) method is the second best in recall. In fact, even for small sets of equality mappings, a significant number of subclass relations can be inferred.

The *Distance-based Polysemic Lexical Comparison* (DPLC) method is the third best method in terms of F-measure. Even if this method has lower precision than the other methods, and its recall is also low (only 48% ontology concepts can be found in WordNet), its contribution is instrumental in improving the overall recall because it finds mappings that the other methods cannot find.

The *Compound Noun Analysis* (CNA) method is the one with the lowest F-measure because of its low recall, which we can explain as follows: the extracted heads from the compounds (most of them are *endocentric*) cannot usually be matched with the target concepts, thus the number of inferred subclass mappings is low. However, this method introduces some mappings that were not found by the other methods (we note that compound names are not in WordNet), at no cost to precision, which is just slightly lower than the precision of the GM and EME methods.

The overall results (All) demonstrates the importance of the combination of the mappings provided by the individual methods. In fact, the overall precision is about the same as that of the most precise method (GM) and the overall recall is much higher than that of each individual method. This phenomenon is similar to what happens in the best "classic" ontology matching system such as AgreementMaker, where the results of several methods, which are targeted to different ontology features (e.g., syntactic, lexical, structural), are combined in a way that significantly improves the final alignment (Cruz et al. 2009b,c).

Discussion. The task of matching LOD ontologies is different from that of matching ontologies in "classic" scenarios, such as those in the OAEI. A major difference relates to the need to have mappings that involve the subclass relation. Another difference is the presence of external concepts that are defined in other ontologies. Therefore, URIs for equivalent concepts will have different prefixes depending on the external ontology. For example, *Person* may appear multiple times with different URIs (e.g., *Person* imported from the *DC* vocabulary and *Person* imported from *DBpedia*). Such differences need to be resolved. A third difference is that LOD ontologies change frequently over time (e.g., addition and removal of concepts or of imported ontologies).

In contrast with the ontology matching tasks in the OAEI, where most of the gold standards involve a 1:1 cardinality constraint, we did not encounter such a restriction for LOD ontologies. If a particular cardinality constraint were imposed, then care would be needed not to lose finer-grained subclass mappings in favor of mappings involving more general classes. In particular, optimization algorithms that determine the set of the final mappings would have to be modified to accommodate such a requirement (Cruz et al. 2009b).

The adoption of external lexical resources such as WordNet and Wikipedia is crucial. The use of such ontologies, and, in our system, of other mediator ontologies, is the reason why BLOOMS and AgreementMaker achieve better results than the other systems. It is hard to find resources that contain most of the needed concepts and hierarchies whose semantics is compliant with the subclass relation. The results show that WordNet has less coverage, but its *hypernym* relation is suitable for this task, while Wikipedia offers more coverage, but the semantics of the *subcategory* relation is less appropriate to derive the semantics of the subclass relation—likely one of the factors that penalizes BLOOMS in precision, as compared with AgreementMaker.

The ontology matching techniques described in this paper raise the same scalability concerns as in classic ontology matching because each source concept is compared to each target concept. In LOD, scalability issues are mostly related to instance matching, not to ontology matching because with few exceptions (e.g., YAGO[9] and OpenCyc)[10] LOD ontologies are not large. Scalability could become more relevant as (1) larger and more complex ontologies are added or (2) real-time matching is needed for "on the go" applications. To address (1), some solutions limit the number of comparisons between concepts in the source and in the target ontologies (Cruz and Sunna 2008). To address (2), specific solutions (albeit outside of the LOD domain) for "on the go" ontology matching have been proposed (Besana and Robertson 2005).

6 Related work

In this section, we discuss related work where schema-level matching is one of the main components (as opposed to instance-level matching Volz et al. 2009). Then we mention an approach that uses background information and two approaches that use lexical annotation methods for schema matching. Finally, we mention three approaches that use the "on the go" paradigm.

The BLOOMS system performs schema-level matching for LOD. It searches Wikipedia pages related to the ontology concepts: the categories extracted from these pages (using a web service) are organized into trees and are compared to support concept matching (Jain et al. 2010). To evaluate its results, BLOOMS uses seven matching tasks, each associated with a pair of popular datasets (e.g., DBpedia, FOAF, GeoNames) and defines the *gold standard* for those tasks. BLOOMS is compared with well-known ontology matching systems such as RiMOM (Li et al. 2009), S-Match (Giunchiglia et al 2007), and AROMA (David et al. 2006), which have participated in the Ontology Alignment Evaluation Initiative (OAEI) (Euzenat et al. 2010). BLOOMS easily outperforms those systems in the LOD domain even if those systems perform better than BLOOMS in the OAEI (Euzenat et al. 2010).

The ontology matching system BLOOMS+, which is an enhanced version of BLOOMS, has been used to align a set of LOD ontologies to the upper level ontology PROTON (Damova et al. 2010). However, PROTON is a well-designed large ontology, more similar to the ontologies considered in the "classic" ontology matching scenarios than in LOD scenarios.

Other systems for LOD include the data fusion tool KnowFuss (Nikolov et al. 2009) and a geospatial linked data tool (Parundekar et al. 2010). The former uses schema-level mappings to improve instance co-reference, but does not address the discovery of schema-level mappings, while the latter is specific to the geospatial domain and infers schema-level mappings from spatial information associated with instances.

The SCARLET system is worth mentioning, even if it has not been evaluated in the LOD domain. It introduces the idea of looking for clues in background web ontologies to assist in the discovery of mappings between two ontologies (Sabou et al. 2008). It searches for concept names in the web ontologies and uses subclass relations defined in those ontologies to derive new mappings. It uses logic-based rules, while our DPLC algorithm uses polysemic

[9] http://www.mpi-inf.mpg.de/yago-naga/yago/.

[10] http://sw.opencyc.org/.

lexical annotations and a probabilistic scoring function. Our GM technique uses URI-based similarity instead of name-based similarity. Our algorithm considers a pool of trusted web ontologies so as to maximize precision.

Next, we describe recent work on the lexical annotation methods of ontology concepts for schema matching and how it has influenced our methods. Our CNA methods is inspired by work on compound names that establishes lexical relationships between terminology concepts (Sorrentino et al. 2010). CNA uses the interpretation of compound names to infer subclass relations between the ontology concepts. Our DPLC algorithm uses word sense disambiguation, following previous work that associates a probability value to semantic relations based on the probability score of the annotations (Po and Sorrentino 2011). In the DPLC algorithm, word sense disambiguation is used to filter the set of annotations. Subclass mappings between concepts (interpreted according to the usual OWL semantics) are then inferred by comparing the set of lexical annotations and computing a polysemic subclass evidence score. We consider that hyponym and subclass relations have different semantics and define the hyponym-to-subclass conversion factor and a distance-driven score.

We briefly point to three ontology matching approaches that specifically address the “on the go” matching paradigm, even if they were not tested in the LOD domain. In one of those approaches, mappings between terms are dynamically discovered during the interaction between autonomous agents and only relevant portions of the ontologies are matched (Besana and Robertson 2005). Another approach matches RDF triples to support semantic interoperability in smart spaces (Smirnov et al. 2010). A third approach proposes a framework where folksonomies are used as mediators in the ontology matching process (Togia et al. 2010).

7 Conclusions

To tap into the huge potential of the LOD cloud, accurate and efficient ontology matching methods are needed. In this paper, we extended the AgreementMaker system, one of the top ontology matching systems in the Ontology Alignment Evaluation Initiative (OAEI) (Cruz et al. 2009c, 2010, 2011b) with four LOD-specific methods: two similarity-based matching methods, namely the *Equivalence Mappings Extension* method and the *Compound Name Analysis* method, and two mediator-based methods, namely the *Distance-based Polysemic Lexical Comparison* method and the *Global Matching* method. A detailed analysis of the contributions of the LOD-specific methods that were added to AgreementMaker shows that each of them plays an instrumental role in the overall result. Furthermore, with these new methods, AgreementMaker amply surpasses the BLOOMS system (Jain et al. 2010), in both precision and F-measure, at the cost of a small penalty in recall.

In the LOD domain, the use of background knowledge to assist in the matching is particularly important. For example, BLOOMS (Jain et al. 2010) uses Wikipedia, and AgreementMaker relies on Wordnet. The fact that Wikipedia provides very good concept “coverage” may be one of the reasons for the 2% advantage in recall of BLOOMS in comparison with AgreementMaker. However, the focus of AgreementMaker on trusted web ontologies (to assist the Global Matching method) and on methods specifically designed to guarantee the accuracy of the mappings, leads to a gain of 16% in precision and 6% in recall when compared with BLOOMS. Finally, run time values show that AgreementMaker significantly outperforms BLOOMS.

The development of accurate and efficient methods for LOD ontology matching provide the basis for the development of “on the go” strategies, where further optimizations can be performed so as to minimize the number of comparisons between concepts in the source and target ontologies, for example by identifying only those parts of the ontologies that need to be matched. This will be the subject of future work.

Acknowledgments Because of the fast evolving nature of the LOD datasets, we thank Prateek Jain for giving us access to the datasets and gold standards as available at the time of the writing of the paper on BLOOMS (Jain et al. 2010).

References

Besana P, Robertson D (2005) Contexts in dynamic ontology mapping. In: AAAI workshop on context and ontology: theory. practice and applications

Bizer C, Heath T, Berners-Lee T (2009) Linked data—the story so far. Int J Semant Web Inf Sys (IJSWIS) 5(3):1–22

Cruz IF, Palandri Antonelli F, Stroe C (2009) AgreementMaker: efficient matching for large real-world schemas and ontologies. PVLDB 2(2):1586–1589

Cruz IF, Palandri Antonelli F, Stroe C (2009b) Efficient selection of mappings and automatic quality-driven combination of matching methods. In: ISWC international workshop on ontology matching (OM) vol 551 of CEUR workshop proceedings, pp 49–60

Cruz IF, Palandri Antonelli F, Stroe C, Keles U, Maduko A (2009c) Using AgreementMaker to align ontologies for OAEI 2009: overview, results, and outlook. In: ISWC international workshop on ontology matching (OM), Vol 551 of CEUR workshop proceedings, pp 135–146

Cruz IF, Palmonari M, Caimi F, Stroe C (2011a) Towards "On the Go" matching of linked open data ontologies. In: IJCAI workshop discovering meaning on the go in large & heterogeneous data (LHD), pp 37–42

Cruz IF, Stroe C, Caci M, Caimi F, Palmonari M, Palandri Antonelli F, Keles UC (2010) Using AgreementMaker to align ontologies for OAEI 2010. In: ISWC international workshop on ontology matching (OM), vol 689 of CEUR workshop proceedings, pp 118–125

Cruz IF, Stroe C, Caimi F, Fabiani A, Pesquita C, Couto FM, Palmonari M (2011b) Using AgreementMaker to align ontologies for OAEI 2011. In: ISWC international workshop on ontology matching (OM), vol 814 of CEUR workshop proceedings, pp 114–121

Cruz IF, Sunna W (2008) Structural alignment methods with applications to geospatial ontologies. Trans GIS Special Issue Semant Similarity Meas Geosp Appl 12(6):683–711

Damova M, Kiryakov A, Simov KI, Petrov S (2010) Mapping the central LOD ontologies to PROTON upper-level ontology. In: ISWC international workshop on ontology matching (OM), vol 689 of CEUR workshop proceedings, pp 61–72

David J, Guillet F, Briand H (2006) Matching directories and OWL ontologies with AROMA. In: International conference on information and knowledge management (CIKM), pp 830–831

Euzenat J, Ferrara A, Hollink L, Isaac A, Joslyn C, Malaisé V, Meilicke C, Nikolov A, Pane J, Sabou M, Scharffe F, Shvaiko P, Spiliopoulos V, Stuckenschmidt H, Sváb-Zamazal O, Svátek V, dos Santos CT, Vouros GA, Wang S (2009) Results of the ontology alignment evaluation initiative 2009. In: ISWC international workshop on ontology matching (OM), vol 551 of CEUR workshop proceedings, pp 73–126

Euzenat J, Ferrara A, Meilicke C, Pane J, Scharffe F, Shvaiko P, Stuckenschmidt H, Sváb-Zamazal O, Svátek V, dos Santos CT (2010) Results of the ontology alignment evaluation initiative 2010. In: ISWC international workshop on ontology matching (OM), vol 689 of CEUR workshop proceedings, pp 85–117

Euzenat J, Ferrara A, van Hage WR, Hollink L, Meilicke C, Nikolov A, Ritze D, Scharffe F, Shvaiko P, Stuckenschmidt H, Sváb-Zamazal O, dos Santos CT (2011) Results of the ontology alignment evaluation initiative 2011. In: ISWC international workshop on ontology matching (OM), vol 814 of CEUR workshop proceedings, pp 85–113

Euzenat J, Shvaiko P (2007) Ontology Matching. Springer, Heidelberg (DE)

Giunchiglia F, Yatskevich M, Shvaiko P (2007) Semantic matching: algorithms and implementation. In: Journal on Data Semantics IX, vol 4601 Lecture Notes in Computer Science. Springer, pp 1–38

Gray AJG, Sadler J, Kit O, Kyzirakos K, Karpathiotakis M, Calbimonte JP, Page K, Garca-Castro R, Frazer A, Galpin I, Fernandes AAA, Paton NW, Corcho O, Koubarakis M, Roure DD, Martinez K, Gmez-Prez A (2011) A semantic sensor web for environmental decision support applications. Sensors 11(9):8855–8887

Jain P, Hitzler P, Sheth A, Verma K, Yeh PZ (2010) Ontology alignment for linked open data. In: International semantic web conference (ISWC), vol 6496 of Lecture Notes in Computer Science. Springer, pp 402–417

Le-Phuoc D, Parreira JX, Hausenblas M, Han Y, Hauswirth M (2010) Live linked open sensor database. In: International conference on semantic systems (I-SEMANTICS). ACM, New York, NY, USA, pp 46:1–46:4

Li J, Tang J, Li Y, Luo Q (2009) RiMOM: a dynamic multistrategy ontology alignment framework. IEEE Trans Data Knowl Eng 21(8):1218–1232

Mendes PN, Jakob M, García-Silva A, Bizer C (2011) DBpedia spotlight: shedding light on the web of documents. In: International conference on semantic systems (I-Semantics)

Nikolov A, Uren V, Motta E, Roeck A (2009) Overcoming schema heterogeneity between linked semantic repositories to improve coreference resolution. In: Asian semantic web conference (ASWC), vol 5926 of Lecture Notes in Computer Science. Springer, pp 332–346

Parundekar R, Knoblock C, Ambite JL (2010) Aligning geospatial ontologies on the linked data web. In: Workshop on linked spatiotemporal data in conjunction with the international conference on geographic information science

Plag I (2003) Word-formation in English. Cambridge University Press,

Po L, Sorrentino S (2011) Automatic generation of probabilistic relationships for improving schema matching. Inf Syst 36(2):192–208

Rahm E, Bernstein PA (2001) A survey of approaches to automatic schema matching. VLDB J 10(4):334–350

Sabou M, d'Aquin M, Motta E (2008) Exploring the semantic web as background knowledge for ontology matching. In: Journal on data semantics XI, vol 5383 of Lecture Notes in Computer Science. Springer, pp 156–190

Smirnov AV, Kashevnik A, Shilov N, Balandin S, Oliver I, Boldyrev S (2010) On-the-fly ontology matching in smart spaces: a multi-model approach. In: International conference on smart spaces and next generation wired/wireless networking (NEW2AN), vol 6294 of Lecture Notes in Computer Science. Springer, pp 72–83

Sorrentino S, Bergamaschi S, Gawinecki M, Po L (2010) Schema label normalization for improving schema matching. Data Knowl Eng 69(12):1254–1273

Staab S, Studer R (eds) (2004) Handbook on ontologies. International handbooks on information systems. Springer, Berlin

Togia T, McNeill F, Bundy A (2010) Harnessing the power of folksonomies for formal ontology matching on-the-fly. In: ISWC international workshop on ontology matching (OM), vol 689 of CEUR workshop proceedings, pp 226–227

Valle ED, Celino I, Dell'Aglio D, Grothmann R, Steinke F, Tresp V (2011) Semantic traffic-aware routing using the larkc platform. IEEE Internet Comput 15(6):15–23

Volz J, Bizer C, Gaedke M, Kobilarov G (2009) Discovering and maintaining links on the web of data. In: International semantic web conference (ISWC), vol 5823 of Lecture Notes in Computer Science. Springer, pp 650–665

Williams E (1981) On the notions "Lexically Related" and "Head of a Word". Linguist Inq 12(2):245–274