Using the Semantic Web as Background Knowledge for Ontology Mapping

Marta Sabou, Mathieu d'Aquin, and Enrico Motta

Knowledge Media Institute (KMi)
The Open University, Milton Keynes, United Kingdom
{r.m.sabou, m.daquin, e.motta}@open.ac.uk

Abstract. While current approaches to ontology mapping produce good results by mainly relying on label and structure based similarity measures, there are several cases in which they fail to discover important mappings. In this paper we describe a novel approach to ontology mapping, which is able to avoid this limitation by using background knowledge. Existing approaches relying on background knowledge typically have one or both of two key limitations: 1) they rely on a manually selected reference ontology; 2) they suffer from the noise introduced by the use of semi-structured sources, such as text corpora. Our technique circumvents these limitations by exploiting the increasing amount of semantic resources available online. As a result, there is no need either for a manually selected reference ontology (the relevant ontologies are dynamically selected from an online ontology repository), or for transforming background knowledge in an ontological form. The promising results from experiments on two real life thesauri indicate both that our approach has a high precision and also that it can find mappings, which are typically missed by existing approaches.
Keywords: ontology mapping, background knowledge, semantic web

1 Introduction

Ontology mapping techniques are essential for building semantic bridges between ontologies. However, current approaches suffer from a number of problems. First of all, most approaches do not provide a formal semantics to the mapping structures they produce[1]. As a result it is difficult for reasoners to make use of these structures, e.g., to answer queries across ontologies [9]. More importantly, current approaches to ontology mapping [11, 13] heavily rely on string-based and structure-based similarity measures. While these techniques can produce good results, there are also numerous examples in which they fail to find mappings.

A few approaches [1, 14, 15] have considered the use of external background knowledge as a way to obtain semantic mappings between syntactically dissimilar ontologies, i.e., to overcome the aforementioned limitations. However, obtaining the right background knowledge is problematic. Some approaches rely on richly

[1] A notable exception is the CtxMatch/S-Match algorithms (see Section 2.1).

axiomatized domain ontologies [1], but unfortunately such ontologies do not exist in all domains and even when they exist, they are unlikely to cover all the intended mappings between the input ontologies. In addition, there are scenarios where it is not possible to select the relevant ontology in advance. For instance, in Semantic Web applications like PowerAqua [9], the domains of the terms to be mapped cannot be determined a priori and whatever background knowledge is needed, must be identified dynamically and in real-time. To avoid the problems associated with the manual selection of an ontology, other techniques try to derive the required background knowledge from weakly structured textual sources [15]. However, given the current limitations in information extraction technology, they then suffer from the resulting noise.

The recent growth of the Semantic Web has resulted in an increased amount of online available semantic data and has led to the first search engine to exploit this data, Swoogle [5]. Our hypothesis is that ontology mapping, while trying to cope with the heterogeneity of the Semantic Web, could actually exploit it. In other words, online available ontologies could provide the background knowledge sources, which are needed to support ontology mapping and to overcome the problems mentioned above. On the one hand, they can be selected dynamically, thus circumventing the need for an a priori, manual ontology selection. On the other hand, by relying on semantic sources, we avoid the inherent noise caused by information extraction based methods.

In this paper we build on these ideas and we describe an approach to ontology mapping which goes beyond similarity-based algorithms by dynamically locating and using relevant background knowledge (Section 3). Since our method derives mappings between pairs of concepts, it can be used to map semantic structures ranging from shallow thesauri to clearly formalized ontologies. We start by discussing in detail the importance of background knowledge in ontology mapping.

2 Motivation

We discuss two major limitations of current ontology mapping approaches that only rely on label and structure similarity (i.e., syntactic approaches), namely that they don't provide semantic mappings (Section 2.1) and that they fail to discover some correct mappings when the mapped ontologies are syntactically dissimilar (Section 2.2). We then point out that while the use of background knowledge can be a solution to these limitations, existing approaches using such knowledge have their shortcomings (Section 2.3).

2.1 Syntactic Approaches Do Not Provide Semantic Mappings

Semantic Web tools, such as PowerAqua, which wish to reason on the results of mapping techniques require that the discovered mappings are expressed as semantic relations between the entities of the ontologies. Formal mapping languages, such as C-OWL [2], envision a wide range of semantic relations that can hold between the entities of two ontologies (e.g., narrower, disjoint). However, few existing mapping techniques are able to discover such semantic mappings.

An analysis of the state of the art of mapping systems presented in [13] explains to some extent the lack of approaches that can provide semantic mappings. The major factor seems to be that most systems combine a range of non-semantic techniques, such as terminological approaches (exploiting string similarity between labels), structural approaches (relying on the structure of the mapped ontologies), and extensional approaches (mapping concepts on the basis of shared instances). Only few systems rely on semantic techniques (also called model based approaches in [13]), thus exploiting the semantics both of the mapped ontologies, and the mapping language, to infer mappings from the available knowledge. As a result, during the last Ontology Alignment Contest (OAC) [6] only one algorithm (CtxMatch [3]) was able to produce partial semantic mappings in the form of subconcept relations. The other techniques produce confidence based mappings that are derived by aggregating the output of terminological and structural algorithms. Unfortunately, this kind of low semantic (*quantitative*) relations are difficult to interpret and to exploit in reasoning procedures. On the contrary, semantic techniques should produce meaningful relations between the mapped entities, on which further reasoning can be applied. They should focus on *qualitatively* good mappings that can be justified and explained through the knowledge and inferences used to deduce them.

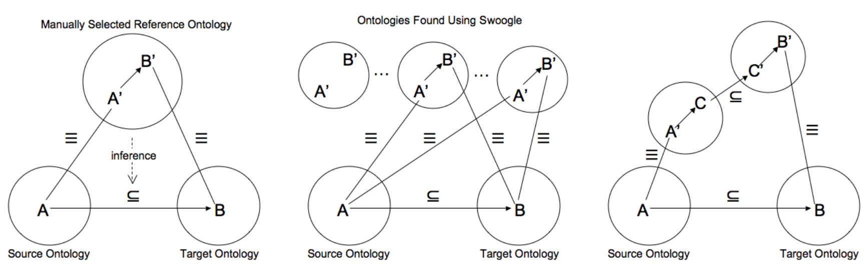
2.2 Syntactic Approaches Fail on Dissimilar Ontologies

As already observed by [1], traditional methods fail when there is little lexical overlap between the labels of the ontology entities, or when the ontologies have weak or dissimilar structures. This observation has been verified to some extent in the last OAC [6]. In the first task of this contest where a base ontology was mapped to its systematically modified versions, the performance of most methods decreased significantly in the test cases where important changes have been performed to the labels and structures of the ontologies (tests 250 - 266). In fact, traditional techniques are based on the hypothesis of an equivalence between some forms of syntactic correspondences and semantic relations. While it is true that, in many cases, string and structural similarities can imply meaningful mappings, this hypothesis is far from being always verified. For instance, the relation between the concepts *Beef* and *Food* may not be discovered on the basis of syntactical considerations, but becomes obvious when considering the meaning of these concepts (their semantics). By ignoring such semantics, syntactic techniques fail to identify several important mappings.

2.3 How is Background Knowledge Currently Used?

The previous sections suggest that the meaning of the mapped concepts should be considered to discover meaningful and syntactically unidentifiable mappings. Unfortunately, while meaning on the Semantic Web is expressed using ontologies, in the case of ontology mapping, the constituents of a mapping can only be given meaning in the context of their own distinct ontology, which cannot cover both the source and target elements, as well as the relation linking them. In other words, a semantic mapping between two ontologies could only be interpreted in a *larger* domain than the ones of these ontologies. Therefore, in order to achieve semantic mapping, the integration of external knowledge is required as a way to cover both input ontologies and to fill the semantic gap between them. So far the following types of background knowledge have been used in mapping:

1. WordNet is one of the most often used sources of background knowledge. For example, CTxMatch [3] (and its follow-up, SMatch [7]) translates ontology labels into logical formulae between their constituents, and maps these constituents to corresponding senses in WordNet. A SAT solver is then used to derive semantic mappings between the different concepts. This approach has been recently extended to handle the problem of *missing background knowledge* [8]. The lack of knowledge is detected and compensated during the mapping process, using techniques still relying on WordNet as a source of knowledge. When using WordNet, it is important to be aware that it is a lexical resource (rather than a truly semantic resource), relating terms by using terminological relations like synonymy or hypernymy. Therefore, it can be seen as a source of linguistic knowledge, useful in relating labels during the terminological step of a matching procedure.



(a) (b) (c)

Fig. 1. Using ontologies as background knowledge for semantic mapping: (a) using a manually selected reference ontology (e.g., [1]); (b) using Swoogle to find the appropriate ontologies (S1) (c) recursively exploiting multiple ontologies (S3).

2. Reference Domain Ontologies. Another approach is to rely on a reference domain ontology as a semantic bridge between two ontologies. In [1], the authors experimentally prove that state of the art matchers fail to satisfactorily match two weakly structured vocabularies of medical terms. As a solution, they propose to use the DICE ontology as a source of background knowledge. Terms from the two vocabularies are first mapped to so called anchor terms in DICE and then their mapping is deduced based on the semantic relation of the anchor terms (see Figure 1(a)). As such, the obtained mappings can describe a larger variety of semantic mappings between terms, not just equivalence. Similarly, [14] presents a case study in the medical domain where mappings between two ontologies are inferred from manually established mappings with a third ontology, and by using the reasoning mechanisms permitted by the C-OWL language.

The advantage of these approaches is that they use richly axiomatized ontologies as background knowledge and therefore guarantee the semantic nature of the mappings. However, a weakness is that the appropriate reference ontology needs to be manually selected prior to mapping. As already pointed out, in many scenarios this approach is unfeasible as we might not know in advance which terms from which ontologies we may want to map. Even in the cases where a reference ontology can be manually selected prior to performing the mapping, there is no guarantee that such an ontology actually exists.

3. Online textual resources can provide an important source of background knowledge. van Hage et. al [15] rely on the combination of two "linguistic ontology mapping techniques" that exploit online available textual sources to resolve mappings between two thesauri in the food domain. On the one hand, they use Google to determine subclass relationships between pairs of concepts using the Hearst pattern based technique introduced by the PANKOW system [4]. On the other hand, they exploit the regularities of an online cooking dictionary to learn hypernym relations between concepts of the source and target ontologies.

The strength of this approach is that it reduces the high cost of establishing adequate background knowledge. Indeed, the background knowledge sources are dynamically discovered and used [15]. There is no need for a manual and domain dependent ontology selection task prior to mapping. The drawback is that the right knowledge has to be extracted first. However, knowledge extraction techniques generally lead to considerable noise and so, reduce the quality of the mapping (e.g., $Mayonnaise \sqsubseteq Cold$). Therefore, without human validation, online texts cannot be considered as reliable semantic resources.

We conclude that the use of background knowledge overcomes the major limitations of syntactic approaches: it allows obtaining semantic relations even between dissimilar ontologies. However, existing approaches either 1) rely on an a priori selected reference ontology or, if they acquire knowledge dynamically, 2) suffer from the noise introduced by knowledge extraction techniques. As a result, they are not suitable for use by novel Semantic Web tools, such as PowerAqua, which require both that the returned mappings are semantically sound and that the relevant background knowledge is dynamically selected, at run-time. In the next section we describe an approach that fulfills these requirements.

3 Using The Semantic Web as Background Knowledge

Our hypothesis is that the growing amount of online available semantic data which makes up the Semantic Web can be used as a source of background knowledge in ontology mapping in a way that satisfies the requirements identified in the previous section. Indeed, this large-scale, heterogeneous semantic data collection provides formally specified knowledge which is likely to be less faulty than that derived from textual sources and therefore lead to better mappings. Moreover, the size and heterogeneity of the collection makes it possible to dynamically select and combine the appropriate knowledge and to avoid the manual selection of a single, large ontology. In the following we investigate increasingly sophisticated approaches to discover and exploit online available ontologies for mapping. We also provide experimental evidence that such mappings can be obtained.

Experimental Data. We have used the dataset described in [15] for our experiments. In their work, van Hage et al. compare the UN FAO's AGROVOC[2] and the USDA Nutrient Database for Standard Reference, release 16 (SR-16)[3] thesauri. Their mapping techniques are verified on a subset of these thesauri. Two modules are selected from AGROVOC one describing food types (A-Food, 21 concepts), the other describing animal products (A-Animal, 88 concepts). These are compared against one module of SR-16 describing meat products (SR-Meat, 24 concepts). Together with the terms (concept names) of these modules, the authors provided us with manually established alignments. The 32 mappings from A-Food to SR-Meat and the 31 mappings from A-Animal to SR-Meat are used here as gold standards for validating the results of our technique.

Implementation Details. We explore our idea by implementing different mapping strategies on top of the Swoogle'05 ontology search engine [5]. Swoogle crawls and indexes a large amount of semantic metadata available online and as such allows access to a large part of the Semantic Web.

Notations. Each strategy takes two candidate concept names (`A` and `B`) as an input and returns the discovered mapping between them. The corresponding concepts in the selected ontology are `A'` and `B'` ("anchor terms"). We rely on the description logic syntax for semantic relations occurring between concepts in an ontology, e.g., $\mathtt{A'} \sqsubseteq \mathtt{B'}$ means that `A'` is a sub-concept of `B'` in a selected ontology and $\mathtt{A'} \perp \mathtt{B'}$ means that `A'` and `B'` are disjoint. The returned mappings are expressed using C-OWL [2] like notations, like $\mathtt{A} \xrightarrow{\sqsubseteq} \mathtt{B}$ or $\mathtt{A} \xrightarrow{\perp} \mathtt{B}$.

3.1 S1: Mappings Based on One Ontology

Our simplest strategy consists in using Swoogle to find ontologies containing concepts with the same names as the candidate concepts and to derive mappings from their relationship in the selected ontologies. Figure 1(b) illustrates this strategy with an example where three ontologies are discovered containing the concepts `A'` and `B'` with the same names as `A` and `B`. The first ontology contains no relation between the anchor concepts, while the other two ontologies contain a subsumption relation. The concrete steps of this strategy are:

1. Select ontologies containing concepts `A'` and `B'` corresponding to `A` and `B`;
2. For each resulting ontology:
   - if $\mathtt{A'} \equiv \mathtt{B'}$ then derive $\mathtt{A} \xrightarrow{\equiv} \mathtt{B}$;
   - if $\mathtt{A'} \sqsubseteq \mathtt{B'}$ then derive $\mathtt{A} \xrightarrow{\sqsubseteq} \mathtt{B}$;
   - if $\mathtt{A'} \sqsupseteq \mathtt{B'}$ then derive $\mathtt{A} \xrightarrow{\sqsupseteq} \mathtt{B}$;

[2] `http://www.fao.org/agrovoc`
[3] `http://www.nal.usda.gov/fnic/foodcomp/Data/SR16/sr16.html`

- if $A' \perp B'$ then derive $A \xrightarrow{\perp} B$;

3. If no ontology is found, no mapping is derived;

Even if this strategy seams simple, it leads to several implementation choices, depending on the relative importance given to time performance and accuracy of the mapping mechanism:

Stop when the first mapping is found. In its simplest version, the algorithm would stop as soon as a mapping is discovered. This is the easiest way to deal with the multiple returned ontologies but it assumes that the first discovered relation can be trusted and there is no need to inspect the other ontologies. Note that the first ontology returned by Swoogle does not necessarily contain a relation between the candidate concepts (like in the example Figure 1(b)). Here we use the first ontology containing such a relation, but, in another implementation, it could be considered that if an ontology covers the candidate concepts without relating them, then no mapping should be derived.

Dealing with contradictions. Instead of relying on the information provided by only one ontology as before, we can envisage to combine the results obtained using all the selected ontologies. Mappings resulting from different sources can be different (e.g., $A \xrightarrow{\sqsubseteq} B$ and $A \xrightarrow{\sqsupseteq} B$), or, in the worst case, inconsistent (e.g., $A \xrightarrow{\sqsubseteq} B$ and $A \xrightarrow{\perp} B$). Several ways of dealing with these contradictions can be considered: we can keep all the mappings (favoring recall), only keep mappings without contradiction (favoring precision), keep the mappings that are derived from most of the ontologies, or try to combine the results (e.g., by deriving $A \xrightarrow{\equiv} B$ from $A \xrightarrow{\sqsubseteq} B$ and $A \xrightarrow{\sqsupseteq} B$). In any case, combining the results from several ontologies is more time consuming (but more reliable) than deriving it from a single ontology.

Considering a particular level of inferences. In the simplest implementation, we can rely on *direct* and *declared* relations between A' and B' in the selected ontology. But, for better results, *indirect* and *inferred* relations should also be exploited (e.g., if $A' \sqsubseteq C$ and $C \perp B'$, then $A' \perp B'$). Different levels of inferences can be considered (no inference, basic transitivity, DL reasoning), each of them representing a particular compromise between the performance of mapping and the completeness of the result.

Experimental results. For our experiments, we implemented this first strategy using basic transitivity reasoning (i.e., taking into account all parents of A' and B') and stopping as soon as a relation was found.

A-Food vs. SR-Meat: We obtained three mappings for these term sets: $Beef, Pork, Poultry \xrightarrow{\sqsubseteq} Food$. All mappings were derived from the Tap ontology[4], where, for example, $Beef \sqsubseteq ReadMeat \sqsubseteq MeatOrPoultry \sqsubseteq Food$.

A-Animal vs. SR-Meat: For these hierarchies, our implementation yielded in a single mapping, $Bacon \xrightarrow{\sqsubseteq} Pork$, which can be found as is in Tap.

Analyzing our results, we discovered that a key factor in the efficiency of our approach is the level to which the candidate terms are covered by Swoogle.

[4] `http://139.91.183.30:9090/RDF/VRP/Examples/tap.rdf`

Indeed, comparing our results to the gold standard mappings, we observed that 24 out of 32 for A-Food vs. SR-Meat (and 20 out of 31 for A-Animal vs. SR-Meat) involve concepts that do not exist in any ontology known to Swoogle (e.g., $GuineaHen, Quail, Squab$). Our experiments are quite strict with respect to finding anchor terms for the candidate concepts: only concepts with identical names are considered. In the next section we suggest ways to reduce this problem.

3.2 S2: Extending Swoogle's Coverage

In order to discover more ontologies that cover the candidate concepts, the process of finding anchor terms must be more flexible. This flexibility can be achieved by considering the following techniques:

A. String normalization. Differences between concept names can be based on simple differences in naming conventions (e.g., $TURKEY\_BREAST$ and $TurkeyBreast$). Most mapping mechanisms use *string normalization* techniques, that consist in transforming strings into a standard form before comparison. Our ontology selection relies on such mechanisms as well.

B. Dealing with compound names. Compound names are particularly difficult to match as they are likely to appear under slightly different forms. Several mapping techniques suggest to be more flexible when searching for compound terms and to allow for:

Different order of the constituents. For example, the term $TurkeyRoast$ does not appear in Swoogle, but *RoastTurkey* does.

Additional constituents. For example, $TurkeyBreast$ is not covered but $TurkeyMeatBreast$ (which additionally contains $Meat$) is.

Less constituents. Some compound terms are only partially covered. For example, $MeatProduct$ does not exist in Swoogle, but $Meat$ does.

Such a flexible matching is also used when discovering anchor terms in the work of Aleksovski et al. [1]. However, while the examples given above are semantically equivalent, automatically identifying lexically different but semantically equivalent compound terms is a difficult task.

B. Exploiting semantic relations between terms. Semantic relations such as synonymy can be used to replace terms with their semantic equivalents. A good source for synonymy information is WordNet. However, the drawbacks of WordNet are that it is difficult to get relevant synonyms unless the sense of the term is known a priory and that compound terms are weakly covered.

Experimental results. Just to prove the point that extended coverage can have a significant effect on the obtained mappings, we rerun our experiments by replacing some terms with their syntactic approximates. We replaced $TurkeyRoast$ with *RoastTurkey* (SR-Meat), $TurkeyBreast$ with $TurkeyMeatBreast$ (SR-Meat) and, $MeatProduct$ with $Meat$ (A-Animal).

A-Food vs. SR-Meat: We obtained that *RoastTurkey* $\xrightarrow{\sqsubseteq}$ $PreparedFood$ and *RoastTurkey* $\xrightarrow{\sqsubseteq}$ *Food* because *RoastTurkey* $\sqsubseteq$ *TurkeyDish* $\sqsubseteq$ *PoultryDish* $\sqsubseteq$ *MeatDish* $\sqsubseteq$ $PreparedFood \sqsubseteq Food$. Also, $TurkeyMeatBreast \xrightarrow{\sqsubseteq}$ *Food*. All mappings were derived from Tap.

A-Animal vs. SR-Meat: We obtained three extra mappings: $Beef \xrightarrow{\sqsubseteq} Meat$[5], $Ham \xrightarrow{\sqsubseteq} Meat$[6] and $Pork \xrightarrow{\sqsubseteq} Meat$[7].

Increasing the coverage of the mapped terms by replacing them with semantically similar variants leads to more mappings. However, another problem comes from the fact that both candidate concepts might not appear in a single ontology, even if each of them appears by itself in many ontologies. Therefore, another way to obtain more mappings is to extend the matching strategy to combine information derived from multiple ontologies, as detailed next.

3.3 S3: Cross-Ontology Mapping Discovery

The previous strategies (S1 and S2) assume that a semantic relation between the candidate concepts can be discovered in a single ontology. However, some relations could be distributed over several ontologies. Therefore, if no ontology is found that relates both candidate concepts, then the mappings should be derived from two (or more) ontologies. In this strategy, mapping is a recursive task where two concepts can be mapped because the concepts they relate in some ontologies are themselves mapped (Figure 1(c)):

1. If no ontologies are found that contain both `A` and `B` then select all ontologies containing a concept `A'` corresponding to `A`;
2. For each of the resulting ontologies:
   (a) for each `C` such that $A' \sqsubseteq C$, search for mappings between `C` and `B`;
   (b) for each `C` such that $A' \sqsupseteq C$, search for mappings between `C` and `B`;
   (c) derive mappings using the following rules:
      - (r1) if $A' \sqsubseteq C$ and $C \xrightarrow{\sqsubseteq} B$ then $A \xrightarrow{\sqsubseteq} B$
      - (r2) if $A' \sqsubseteq C$ and $C \xrightarrow{\equiv} B$ then $A \xrightarrow{\sqsubseteq} B$
      - (r3) if $A' \sqsubseteq C$ and $C \xrightarrow{\perp} B$ then $A \xrightarrow{\perp} B$
      - (r4) if $A' \sqsupseteq C$ and $C \xrightarrow{\sqsupseteq} B$ then $A \xrightarrow{\sqsupseteq} B$
      - (r5) if $A' \sqsupseteq C$ and $C \xrightarrow{\equiv} B$ then $A \xrightarrow{\sqsupseteq} B$

In this strategy, steps (a) and (b) can be ran in parallel and stopped when one of them is able to establish a mapping. These two steps correspond to the recursive part of the algorithm. The task of *searching for mappings between C and B* can be realized using one of our three strategies.

Experimental results. We have implemented this algorithm by using the first mapping strategy (S1) in the recursive part.

A-Food vs. SR-Meat: By combining information available in different ontologies, we obtained that $Chicken, Duck, Goose, Turkey \xrightarrow{\sqsubseteq} Food$ because they are subclasses of *Poultry* in some ontologies [8] and $Poultry \sqsubseteq Food$ in

[5] `http://reliant.teknowledge.com/DAML/Mid-level-ontology.daml`
[6] `http://www.pizza-to-go.org/ontology`
[7] `http://reliant.teknowledge.com/DAML/Economy.daml`
[8] e.g., `http://reliant.teknowledge.com/DAML/Mid-level-ontology.daml`

Tap (r1). We also discovered that $Ham \xrightarrow{\sqsubseteq} Food$ because $Ham \sqsubseteq Meat$ and $Meat \sqsubseteq Food$ in SUMO[9] (r1). Finally, we found that $Ham \xrightarrow{\perp} Seefood$ because $Ham \sqsubseteq Meat$ and $Meat \perp Seafood$[10] (r3).

A-Animal vs. SR-Meat: Because $Beef, Ham, Pork \sqsubseteq Meat$ and $Meat \perp Seafood$ we derive that $Beef, Ham, Pork \xrightarrow{\perp} Seafood$ (r3). Note that incompatibility mappings are not specified in the gold standard.

4 Conclusions

The aim of this paper was to show the feasibility and the potential advantages of using automatically selected online ontologies as background knowledge for semantic mapping. As Table 1 shows, our experiments on two real life examples have provided promising results, which are consistent with our idea of semantic mappings, as discussed in Section 2. In particular, the output of our algorithm provides mappings, which i) are expressed in terms of semantic relations (subsumption, disjunction); ii) rely on semantics, as expressed in external ontologies; and iii) in many cases would have not been discovered by syntactic techniques (e.g., because the strings denoting similar concepts are very different).

	Mappings	
	A-Food vs. SR-16	**A-Animal vs. SR-16**
S1	$Beef, Pork, Poultry \xrightarrow{\sqsubseteq} Food$	$Bacon \xrightarrow{\sqsubseteq} Pork$
S2	$RoastTurkey \xrightarrow{\sqsubseteq} Food, PreparedFood$ $TurkeyMeatBreast \xrightarrow{\sqsubseteq} Food$	$Beef, Ham, Pork \xrightarrow{\sqsubseteq} Meat$
S3	$Chicken, Goose, Turkey, Duck \xrightarrow{\sqsubseteq} Food$ $Ham \xrightarrow{\sqsubseteq} Food$; $Ham \xrightarrow{\perp} Seefood$*	$Beef, Ham, Pork \xrightarrow{\perp} Seafood$*
Total	11 (+1*)	4 (+3*)

Table 1. Discovered mappings. Marked mappings* do not exist in the Gold Standard.

Note that the technique presented here is not meant to be used in isolation, as an alternative to current approaches. On the contrary, we plan to integrate our technique with "syntactic" techniques, to develop a robust and comprehensive ontology mapping method. For this reason it is difficult to provide a detailed comparison with other approaches, using standard measures of precision and recall. Because our technique is meant to enhance, rather than replace existing methods, it scores very highly on precision but relatively low on recall, about 30% on the test case provided by [15] (low recall is due to the fact that many concepts are not covered by any online ontology). Having said so, we should also emphasize that some of the mappings our system is able to discover are not even covered by the Gold Standard defined in [15], which is an indication of the greater range of mapping possibilities provided by our approach.

[9] `http://reliant.teknowledge.com/DAML/SUMO.owl`

[10] `http://ontolingua.stanford.edu/doc/chimaera/ontologies/wines.daml`

The broader context of our work is one of providing meaningful mappings that can be used by the next generation of Semantic Web applications [10] to reason over multiple ontologies. Hence, in contrast with most existing work on ontology mapping, we are interested in developing an approach which can be used by systems that need to create mappings dynamically and in real time to make use of the large scale semantics available on the Web. This aspect still needs to be evaluated using appropriate experimental settings and criteria.

Another goal of this paper was to identify some of the research issues brought up by the innovative aspects of our technique. The first innovative aspect is that appropriate background knowledge is automatically selected from the variety of ontologies available on the Semantic Web. As a result, important issues for us concern i) the current level of semantic coverage of the Semantic Web and ii) the quality of the tools that give access to it. Regarding the second point, although Swoogle [5] is by far the most advanced ontology search engine available today, it is still rather limited with respect to supporting our needs to exploit online ontologies dynamically and in real time. Among other things, we need better query facilities, a richer set of relations between ontologies (at the very least to quickly discard duplicate ontologies), and other ranking mechanisms in addition to popularity. Regarding the current level of coverage on the Semantic Web, our previous work [12] indicated a *knowledge sparseness phenomenon*: some domains are well covered by existing ontologies (e.g., academic research and medicine), while others are not covered at all. This phenomenon has a direct influence on our method as coverage of a domain is a prerequisite for successfully mapping ontologies in this domain. Having said so, there is evidence that the Semantic Web is rapidly growing and, as a result, our method will be able to perform better and better, simply by taking advantage of the improved semantic coverage.

The second innovative aspect of our technique, that of combining facts from different ontologies, leads to another important issue: how to deal with contradictions. Online ontologies are made for different purposes, in different contexts and therefore, can lead to contradictory (or inconsistent) mappings. As already mentioned, one of the advantages of semantic techniques is that resulting mappings can be justified and explained. In that sense, one way to deal with contradictions would be to relate mappings to the ontologies on which they are based. Using this solution, contradictory mappings would still co-exist but in a *contextualized* form, i.e., justified and valid only in the context of some particular ontologies. Another way to deal with contradictions would be to rely only on ontologies sharing a similar context with the mapped ones. Indeed, when trying to map the $Turkey$ concept, we would more likely find relevant mappings using ontologies also containing concepts like $Food$ or $Meat$, than ontologies covering countries. This implies that more advanced ontology selection techniques are needed which can consider similarity between ontologies as a selection criterion.

In addition to tackling the aforementioned issues, the experiments presented here need to be followed by several studies. A key next step concerns the complete implementation and evaluation of our technique, which is currently restricted to subsumption and disjunction relations between concepts. Our plan is to extend the technique to also map properties and individuals, so to increase the range of discovered mappings. Finally, when presenting our strategies we emphasized the trade-off between the performance and the accuracy of the mapping mechanism. Finding a good compromise between these two aspects is a hard task and we plan to address this issue by reformulating our technique as an *anytime* algorithm.

Acknowledgements

We are grateful to Willem van Hage for providing us with his experimental data set. This work was funded by the AKT project sponsored under grant number GR/N15764/01, and the Open Knowledge and NeOn projects sponsored under EC grant numbers IST-FF6-027253 and IST-FF6-027595.

References

1. Z. Aleksovski, M. Klein, W. ten Katen, and F. van Harmelen. Matching Unstructured Vocabularies using a Background Ontology. In S. Staab and V. Svatek, editors, *Proc. of EKAW*, LNAI. Springer-Verlag, 2006.
2. P. Bouquet, F. Giunchiglia, F. van Harmelen, L. Serafini, and H. Stuckenschmidt. Contextualizing ontologies. *Journal of Web Semantics*, 1(4):24, 2004.
3. P. Bouquet, L. Serafini, and S. Zanobini. Peer-to-Peer Semantic Coordination. *Journal of Web Semantics*, 2(1), 2005.
4. P. Cimiano, S. Handschuh, and S. Staab. Towards the Self-Annotating Web. In *Proc. of the 13th International World Wide Web Conference*, 2004.
5. L. Ding, R. Pan, T. Finin, A. Joshi, Y. Peng, and P. Kolari. Finding and Ranking Knowledge on the Semantic Web. In *Proc. of ISWC*, 2005.
6. J. Euzenat, H. Stuckenschmidt, and M. Yatskevich. Introduction to the Ontology Alignment Evaluation 2005. In *Proc. of the Integrating Ontologie WS*, 2005.
7. F. Giunchiglia, P. Shvaiko, and M. Yatskevich:. Semantic Schema Matching. In *Proc. of CoopIS'05*, volume 3760 of *LNCS*, pages 347 – 360, 2005.
8. F. Giunchiglia, P. Shvaiko, and M. Yatskevich. Discovering Missing Background Knowledge in Ontology Matching. In *Proc. of ECAI*, 2006.
9. V. Lopez, E. Motta, and V. Uren. PowerAqua: Fishing the Semantic Web. In *Proc. of ESWC*, 2006.
10. E. Motta and M. Sabou. Next Generation Semantic Web Applications. In *Proc. of the 1st Asian Semantic Web Conference (ASWC)*, 2006.
11. N. Noy. Semantic Integration: A Survey of Ontology-based Approaches. *SIGMOD Record*, 33(4), 2004.
12. M. Sabou, V. Lopez, and E. Motta. Ontology Selection on the Real Semantic Web: How to Cover the Queens Birthday Dinner? In *Proc. of EKAW*, 2006.
13. P. Shvaiko and J. Euzenat. A Survey of Schema-based Matching Approaches. *Journal on Data Semantics*, IV, 2005.
14. H. Stuckenschmidt, F. van Harmelen, L. Serafini, P. Bouquet, and F. Giunchiglia. Using C-OWL for the Alignment and Merging of Medical Ontologies. In *Proc. of the First Int. WS. on Formal Biomedical K. R. (KRMed)*, 2004.
15. W. van Hage, S. Katrenko, and G. Schreiber. A Method to Combine Linguistic Ontology-Mapping Techniques. In *Proc. of ISWC*, 2005.