A Framework for Aligning Ontologies

Patrick Lambrix and He Tan

Department of Computer and Information Science,
Linköpings universitet, Sweden
{patla, hetan}@ida.liu.se

Abstract. Ontologies are an important technology for the Semantic Web. In different areas ontologies have already been developed and many of these ontologies contain overlapping information. Often we would therefore want to be able to use multiple ontologies and thus the ontologies need to be aligned. Currently, there exist a number of systems that support users in aligning ontologies, but not many comparative evaluations have been performed.

In this paper we present a general framework for aligning ontologies where different alignment strategies can be combined. Further, we exemplify the use of the framework by describing a system (SAMBO) that is developed according to this framework. Within this system we have implemented some already existing alignment algorithms as well as some new algorithms. We also show how the framework can be used to experiment with combinations of strategies. This is a first step towards defining a framework that can be used for comparative evaluations of alignment strategies. For our tests we used several well-known bio-ontologies.

1 Introduction

Intuitively, ontologies (e.g. [10,6]) can be seen as defining the basic terms and relations of a domain of interest, as well as the rules for combining these terms and relations. They are considered to be an important technology for the Semantic Web. Ontologies are used for communication between people and organizations by providing a common terminology over a domain. They provide the basis for interoperability between systems. They can be used for making the content in information sources explicit and serve as an index to a repository of information. Further, they can be used as a basis for integration of information sources and as a query model for information sources. They also support clearly separating domain knowledge from application-based knowledge as well as validation of data sources. The benefits of using ontologies include reuse, sharing and portability of knowledge across platforms, and improved maintainability, documentation, maintenance, and reliability. Overall, ontologies lead to a better understanding of a field and to more effective and efficient handling of information in that field. In the field of bioinformatics, for instance, the work on ontologies is recognized as essential in some of the grand challenges of genomics research [1] and there is much international research cooperation for the development of ontologies (e.g. the Gene Ontology (GO) [5] and Open Biomedical Ontologies (OBO) [19]

efforts) and the use of ontologies for the Semantic Web (e.g. the EU Network of Excellence REWERSE [23,24]).

Many ontologies have already been developed and many of these ontologies contain overlapping information. Often we would therefore want to be able to use multiple ontologies. For instance, companies may want to use community standard ontologies and use them together with company-specific ontologies. Applications may need to use ontologies from different areas or from different views on one area. Ontology builders may want to use already existing ontologies as the basis for the creation of new ontologies by extending the existing ontologies or by combining knowledge from different smaller ontologies. In each of these cases it is important to know the relationships between the terms in the different ontologies. We say that we align two ontologies when we define the relations between terms in the different ontologies. We merge two ontologies when we, based on the alignment relations between the ontologies, create a new ontology containing the knowledge included in the source ontologies.

Ontology alignment and merging is recognized as an important step in ontology engineering that needs more extensive research (e.g. [20]). Currently, there exist a number of systems that support users in merging or aligning ontologies in the same domain. These systems use different techniques, but it is not clear how well these techniques perform for different types of ontologies. Few comparative evaluations on ontology merging and alignment have been performed [12,13,20] and no tools for supporting these kinds of evaluations exist yet [8].

In this paper we propose a framework for aligning ontologies. We identify different types of strategies (section 3.1) and show how these strategies can be integrated in one framework (section 3.2). Further, we exemplify the use of the framework by describing a system (SAMBO) that is developed according to this framework (section 4). Within this system we have implemented some already existing alignment algorithms as well as some new algorithms. We also show how the framework can be used to combine different strategies and to experiment with these combinations. This is a first step towards defining a framework that can be used for comparative evaluations of alignment and merging strategies. We tested different combinations of alignment algorithms on several bio-ontologies and discuss the results in section 5. In the next section we provide some background on (bio-)ontologies and ontology alignment systems.

2   Background

2.1   Ontologies

Ontologies differ regarding the kind of information they can represent. From a knowledge representation point of view ontologies can have the following components (e.g. [10,26]). Concepts represent sets or classes of entities in a domain. Instances represent the actual entities. They are, however, often not represented in ontologies. Further, there are many types of relations. Finally, axioms represent facts that are always true in the topic area of the ontology. These can be

such things as domain restrictions, cardinality restrictions or disjointness restrictions. Depending on which of the components are represented and the kind of information that can be represented, we can distinguish between different kinds of ontologies such as controlled vocabularies, taxonomies, thesauri, data models, frame-based ontologies and knowledge-based ontologies. These different types of ontologies can be represented in a spectrum of representation formalisms ranging from very informal to strictly formal. For instance, some of the most expressive representation formalisms in use for ontologies are description logic-based languages such as DAML+OIL and OWL.

2.2   Bio-ontologies

In this paper we have chosen to use test cases based on bio-ontologies (e.g. [10]). There are several reasons for this. Research in bio-ontologies is recognized as essential in some of the grand challenges of genomics research [1]. The field has also matured enough to develop standardization efforts. An example of this is the organization of the first conference on Standards and Ontologies for Functional Genomics (SOFG) in 2002 and the development of the SOFG resource on ontologies. Further, there exist ontologies that have reached the status of de facto standard and are being used extensively for annotation of databases. Also, OBO was started as an umbrella web address for ontologies for use within the genomics and proteomics domains. Many bio-ontologies are already available via OBO. There are also many overlapping ontologies available in the field.

The ontologies that we use in this paper are GO ontologies, Signal-Ontology (SigO), Medical Subject Headings (MeSH) and the Anatomical Dictionary for the Adult Mouse (MA). The GO Consortium is a joint project which goal is to produce a structured, precisely defined, common and dynamic controlled vocabulary that describes the roles of genes and proteins in all organisms. Currently, there are three independent ontologies publicly available over the Internet: biological process, molecular function and cellular component. The GO ontologies are a de facto standard and many different bio-databases are today annotated with GO terms. The terms in GO are arranged as nodes in a directed acyclic graph, where multiple inheritance is allowed. The purpose of the SigO project is to extract common features of cell signaling in the model organisms, try to understand what cell signaling is and how cell signaling systems can be modeled. SigO is a publicly available controlled vocabulary of the cell signaling system. It is based on the knowledge of the Cell Signaling Networks data source [30] and treats complex knowledge of living cells such as pathways, networks and causal relationships among molecules. The ontology consists of a flow diagram of signal transduction and a conceptual hierarchy of biochemical attributes of signaling molecules. MeSH is a controlled vocabulary produced by the American National Library of Medicine and used for indexing, cataloguing, and searching for biomedical and health-related information and documents. It consists of sets of terms naming descriptors in a hierarchical structure. These descriptors are organized in 15 categories, such as the category for anatomic terms, which is the category we use in the evaluation. MA is cooperating with the Anatomical Dictionary for Mouse Development

(EMAP) to generate an anatomy ontology (controlled vocabulary) covering the entire lifespan of the laboratory mouse. It organizes anatomical structures spatially and functionally, using is-a and part-of relationships.

2.3   Ontology Alignment and Merging Systems

There exist a number of ontology alignment systems that support the user to find inter-ontology relationships. Some of these systems are also ontology merging systems. However, up to date only two comparative evaluations of ontology merge systems have been performed. The EU OntoWeb project [20] evaluated the systems PROMPT [16] based on Protégé (with extension Anchor-PROMPT [17]), Chimaera [14] (described, not evaluated), FCA-Merge [28] and ODEMerge. This evaluation focused on such things as functionality, interoperability and visualization, but did not include tests on the quality of the alignment. In [12,13] PROMPT, Chimaera and a previous version of SAMBO were evaluated in terms of the quality of the alignment as well as the time it takes to align ontologies with these tools. There are other tools such as ArtGen [15], ASCO [11], GLUE [2], HCONE [9], IF-Map [33], iMapper [27], ITTalks [29], QOM [3], and S-Match [7], but these have not appeared in comparative evaluation studies. For the sake of brevity, we do not describe the different systems in detail, but we show a summary of the strategies that are used by the systems in table 1 in the next section.

3   Framework for Ontology Alignment

In this section we introduce a framework for ontology alignment. We describe different strategies for alignment and show they can be integrated in a general framework. Although we focus on alignment, we also briefly show how the framework can be extended to also cover ontology merging.

3.1   Strategies

Different strategies are based on different kinds of knowledge that can be exploited during the alignment process to enhance the effectiveness and efficiency. Some of the approaches use information inherent in the ontologies. Other approaches require the use of external sources. We describe currently used strategies and in table 1 we give an overview of the used strategies per system.

– *Strategies based on linguistic matching.* These approaches make use of textual descriptions of the concepts and relations such as names, synonyms and definitions. The similarity measure between concepts is based on comparisons of the textual descriptions. Simple string matching approaches and information retrieval approaches (e.g. based on frequency counting) may be used. Most systems use this kind of strategies.
– *Structure-based strategies.* These approaches use the structure of the ontologies to provide suggestions. Typically, a graph structure over the concepts is provided through is-a, part-of or other relations. The similarity of concepts

Table 1. Strategies used by alignment systems

	linguistic	structure	constraints	instances	auxiliary
ArtGen	name	parents, children		domain-specific documents	WordNet
ASCO	name, label description	parents, children, siblings, path from root			WordNet
Chimaera	name	parents, children			
FCA-Merge	name			domain-specific documents	
GLUE	name	neighborhood		instances	
HCONE	name	parents, children			WordNet
IF-Map				instances	a reference ontology
iMapper		leaf, non-leaf, children, related node	domain, range	documents	WordNet
ITTalks		parents, children		documents	
(Anchor-) PROMPT	name	direct graphs			
QOM	name label	parents, children	equivalence		
SAMBO	name, synonym	is-a and part-of, descendants and ancestors		documents	WordNet, UMLS
S-Match	label	path from root	semantic relations codified in labels		WordNet

is based on their environment. An environment can be defined in different ways. For instance, using the is-a relation (e.g. [13]) an environment could be defined using the parents (or ancestors) and the children (or descendants) of a concept. Some approaches also use other relations (e.g. [17]).

– *Constraint-based approaches.* In this case the axioms are used to provide suggestions. For instance, knowing that the range and domain of two relations are the same, may be an indication that there is a relationship between the relations. Similarly, when two concepts are both disjoint with a third concept, we may have a similarity between the first two concepts. On their own these approaches may not be sufficient to provide high quality suggestions, but they may complement other approaches to reduce the number of irrelevant suggestions. Constraint-based approaches are currently used by only a few systems.
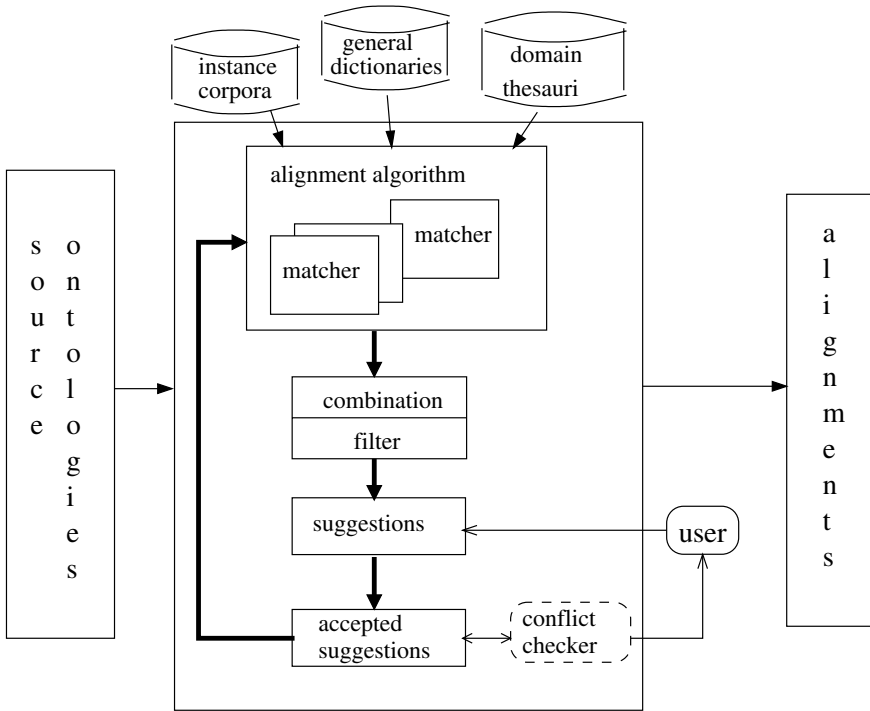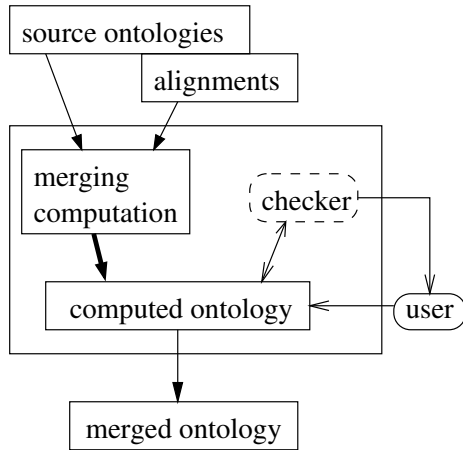
Fig. 1. A general alignment strategy

Fig. 2. A general merging algorithm

- *Instance-based strategies.* In some cases instances are available directly or can be obtained. For instance, the entries in biological databases that are annotated with GO terms, can be seen as instances for these GO terms. When

instances are available, they may be used in defining similarities between concepts.

- *Use of auxiliary information.* Dictionaries and thesauri representing general or domain knowledge, or intermediate ontologies may be used to enhance the alignment process. They provide external resources to interpret the intended meaning of the concepts and relations in an ontology (e.g. [15]). Also information about previously merged ontologies may be used. Many systems use auxiliary information.
- *Combining different approaches.* The different approaches use different strategies to compute similarity between concepts. Therefore, a combined approach may give better results. Although most systems combine different approaches, not much research is done on the applicability and performance of these combinations.

3.2   Framework

In figure 1 we propose a general alignment framework. An alignment algorithm receives as input two source ontologies. The algorithm can include several matchers. Each matcher utilizes knowledge from one or multiple sources. For instance, a linguistic matcher uses textual descriptions and may use auxiliary information in the form of a general dictionary. The matchers calculate similarities between the concepts and relations from the different source ontologies. Alignment suggestions are then determined by combining and filtering the results generated by one or more matchers. For instance, similarity results from a linguistic matcher and a learning matcher may be combined and the pairs of concepts and relations with a similarity value above a certain threshold are retained as alignment suggestions. By using different matchers and combining them and filtering in different ways we obtain different alignment strategies. The suggestions are presented to the user who accepts or rejects the suggestions. The acceptance and rejection of a suggestion may influence further suggestions. Also, some matchers (e.g. some structural matchers as in [17,13]) require as input already accepted suggestions. Further, a conflict checker is used to avoid conflicts introduced by alignment relationships. The output of the alignment algorithm is a set of alignment relations between concepts and relations from the source ontologies.

Figure 2 shows a simple merging algorithm. A new ontology is computed from the source ontologies and their identified alignment. The checker is used to avoid conflicts as well as to detect unsatisfiable concepts and, if so desired by the user, to remove redundancy.

4   SAMBO

In this section we describe a prototype of SAMBO, an ontology alignment and merging tool implemented according to the framework described in section 3.

4.1   System

The current implementation of SAMBO is a web-based system that helps a user to merge two ontologies into a new ontology with unique names for terms. In

this implementation the system supports ontologies in OWL and DAML+OIL formats. After loading the source ontologies, the user can start the alignment process. The system separates the process into two steps: aligning relations and aligning concepts. The second step can be started after the first step is finished. In each step, the user can choose to manually merge terms (i.e. equivalent terms) or introduce is-a relationships in the ontologies, or to have the system propose suggestions. The user can choose to accept or reject the suggestions. Upon an action of the user, the suggestion list is updated. If the user rejects a suggestion where two different terms have the same name, she is required to rename one of the terms. At each point in time the user can view the ontologies represented in trees with the information on which actions have been performed, and she can check how many suggestions there are left for the step. After the user accomplishes the alignment process, the system receives the final alignment list and can be asked to create the new ontology. The system merges the terms in the alignment list, computes the consequences, makes the additional changes that follow from the operations, and finally copies the other terms to the new ontology. Furthermore, SAMBO uses a DIG description logic reasoner (e.g. Racer [25], FaCT [4]) to provide a number of reasoning services. The user can ask the system whether the new ontology is consistent and can ask for information about unsatisfiable concepts and cycles in the ontology.

4.2   Alignment Algorithm

For this implementation of SAMBO we experimented with the combination of already existing strategies as well as some newly implemented strategies.

The *terminological matcher* contains matching algorithms based on the textual descriptions (names and synonyms) of concepts and relations. In the current implementation, the matcher includes two approximate string matching algorithms, n-gram and edit distance, and a linguistic algorithm. A n-gram is a set of n consecutive characters extracted from a string. Similar strings will have a high proportion of n-grams in common. Edit distance is defined as the number of deletions, insertions, or substitutions required to transforming one string into the other. The greater the edit distance, the more different the strings are. The linguistic algorithm computes the similarity of the terms by comparing the lists of words of which the terms are composed. Similar terms have high proportion of words in common in the lists. A porter stemming algorithm is employed to each word. Further, a general thesaurus, WordNet [32], is used to enhance the similarity measure by looking up the hypernym relationships of the pairs of words in WordNet. All these matchers were evaluated in [13]. The terminological matcher outputs similarity values by combining the results from these three algorithms using weights. The similarity between a concept C_1 from the first source ontology and a concept C_2 from the second source ontology is defined as $tsim(C_1, C_2) = \sum_{k=1}^{3} w_k * tsim_k(C_1, C_2)$ where w_k are the weights assigned to the algorithms. If the weights are chosen carefully, this combination can overcome the weaknesses of the individual algorithms. In our experiments

we used the weights 0.3, 0.3 and 0.21 for the linguistic algorithm, edit distance and n-gram, respectively.

The *structural matcher* is an iterative algorithm based on the *is-a* and *part-of* hierarchies of the ontologies (table 2). The algorithm requires as input a list of alignment relations and can therefore not be used in isolation. The intuition behind the algorithm is that if two concepts lie in similar positions in the two hierarchies, then probably they are similar. The propagation coefficients indicate how well the similarity of a given alignment propagates to its neighbors, and ranges from 0 to 1. Similarity of identified alignments propagates to their ancestors and descendants, but the effect diminishes with respect to distance, and therefore a maximal distance can be set optionally.

Table 2. The structure-based algorithm

$//MatchSet(O_1, O_2)$ is the set of identified alignment relations.
for each element (c_{1i}, c_{2j}) **in** $MatchSet(O_1, O_2)$
  for all ancestors c_{1m} **of** c_{1i}
    for all ancestors c_{2n} **of** c_{2j}
    $//c_p$ is the propagation coefficient for the *is-a/part-whole* relation
    $//l(c_{1i}, c_{1m})$ is the length of the path between c_{1i} and c_{1m}
    $ssim(c_{1m}, c_{2n}) = \frac{c_p}{l(c_{1i}, c_{1m})} * \frac{c_p}{l(c_{2j}, c_{2n})}$
  for all descendants c_{1p} **of** c_{1i}
    for all descendants c_{2q} **of** c_{2j}
    $//c_c$ is the propagation coefficient for the *inverse-is-a/whole-part* relation
    $ssim(c_{1p}, c_{2q}) = \frac{c_c}{l(c_{1i}, c_{1p})} * \frac{c_c}{l(c_{2j}, c_{2q})}$

Another strategy is to use *domain knowledge*. We utilize the domain lexicon Unified Medical Language System (UMLS) [31], a repository of biomedical vocabularies. The similarity of two terms in the source ontologies is determined by how they are mapped to terms in UMLS.

We also included a *learning matcher*. We created a corpus containing documents extracted from PubMed [21]. To each term in the source ontologies we assign at most 100 documents, which are the abstracts of the articles retrieved from PubMed using the name of the term as the search term. The similarity value between a concept C_1 in the first source ontology and concept C_2 in the second source ontology is computed as $lsim(C_1, C_2) = [P(C_1|C_2) + P(C_2|C_1)]/2$, where the probability $P(C_1|C_2)$ is estimated to be the fraction of the total number of documents associated with C_2 that are also classified to be associated with C_1, and similarly for $P(C_2|C_1)$. A naive Bayes classifier is applied to classify documents.

The matchers compute similarity values in [0..1], where 1 indicates an expected similarity. The suggestions proposed to the user are those whose similarity values are higher than a threshold (filter in figure 1). The user is given the choice to employ one or several matchers during the alignment process. The suggestions can be determined based on the similarity value from one matcher,

or the combination of the similarity values measured by several matchers using weights, $sim(C_1, C_2) = (\sum_{k=1}^{n} w_k * sim_k(C_1, C_2))/n$, where n is the number of combined matchers and sim_k and w_k represent the similarity values and weights, respectively, for the different matchers (combination in figure 1).

5   Evaluation

In the evaluation we compare the quality of the alignment suggestions that are generated by our different matchers and their combinations.

5.1   Test Cases

We created five test cases based on two groups of bio-ontologies. For the first two cases we use a part of a GO ontology together with a part of SigO. Each case was chosen in such a way that there was an overlap between the GO part and the SigO part. The first case, *behavior* (B), contains 57 terms from GO and approximately 10 terms from SigO. The second case, *immune defense* (ID), contains 73 terms from GO and 15 terms from SigO. We used more terms from GO than from SigO because the granularity of GO is higher than the granularity of SigO for these topics.

The other cases are taken from two bio-ontologies that are available from OBO: MeSH (anatomy category) and MA. The two ontologies cover a similar subject domain, anatomy, and are developed independently. The three cases used in our test are: *nose* (containing 15 terms from MeSH and 18 terms from MA), *ear* (containing 39 terms from MeSH and 77 terms from MA), and *eye* (containing 45 terms from MeSH and 112 terms from MA). We translated the ontologies from the GO flat file format to OWL retaining identifiers, names, synonyms, definitions and is-a and part-of relationships.

5.2   Comparison of Matchers

We compare the quality of the suggestions that are generated by the different matchers and their combinations.

In table 3 we present information about the suggestions generated by the individual matchers: terminological, terminological using WordNet, algorithm using domain knowledge (UMLS), and learning. The cases are given in the first column. The second column represents the number of expected suggestions. For instance, in the 'ear' case, there are 29 alignments that are specified by domain experts. This is the minimal set of suggestions that matchers are expected to generate for a perfect recall. This set does not include the inferred suggestions. Inferred suggestions will be inferred by the merging algorithm and we therefore consider them neither as correct nor as wrong suggestions. An example of an inferred suggestion is that incus is a kind of ear ossicle. In this case we know that auditory bone (MA) is the same as ear ossicle (MeSH), and incus is a kind of auditory bone in MA. Then the system should derive that incus is a kind of ear ossicle. The learning matcher (last column) generates 14 suggestions of which

Table 3. Comparison of algorithms

Case	E.Sugs	Terminological	T.+ WordNet	Domain	Learning
B	4	4/4/0	4/4/0	4/4/0	4/4/0
ID	8	6/4/2	6/4/2	4/4/0	5/5/0
nose	7	6/6/0	6/6/0	7/7/0	5/5/0
ear	29	26/26/0	27/27/0	25/25/0	14/14/0
eye	28	21/21/0	22/21/1	21/21/0	19/16/3

14 suggestions are correct and no suggestion is wrong. The structural matcher requires a set of already identified alignments as input, and thus there are no results for the structural matcher in table 3.

The test ontologies provide a lot of synonyms and therefore the quality of the suggestions from the terminological matcher is good. The matcher finds suggestions where the names of terms are completely different, e.g. (inner ear, labyrinth), where inner ear has labyrinth as synonym. The matcher also gives suggestions where the names of terms are slightly different, e.g. (stapes, stape). By using a general dictionary (WordNet), it finds suggestions such as (perilymphatic channel, cochlear aqueduct) where cochlear aqueduct has perilymphatic duct as synonym, and duct is a synonym of channel in WordNet. On the other hand, since endothelium is a kind of epithelium in WordNet, it generates a wrong suggestion (corneal endothelium, corneal epithelium). The quality of the suggestions from the domain matcher is also good. The matcher finds suggestions of which the terms have completely different names and synonyms, or have no synonyms at all, e.g. (external acoustic meatus, ear canal). The matcher works for some terms with slightly different names, e.g. (optic disc, optic disk), which are mapped to the concept optic disc in UMLS, but does not work on others, e.g. (stapes, stape), which are mapped to different concepts in UMLS. The quality of the suggestions from the learning matcher varies in the different ontologies in this evaluation. In the 'ID' case it produces the best result among the matchers. It avoids the wrong suggestions with slightly different names, such as (B cell activation, T Cell Activation). It also finds the suggestion (natural killer cell activation, Natural Killer Cell Response), which is not found by other matchers. However, in the 'eye' case it produces the worst result. In this case all its correct suggestions are also found by the other matchers. The quality of the suggestions from the learning matcher depends on the associated documents retrieved from PubMed. One factor that plays a role in this is that the terms are used as search strings and thus may appear anywhere in the documents. Another factor is that for some concepts only few documents are retrieved.

Table 4 shows the quality of the *extra* suggestions generated by the structural matcher based on the alignment results given by the other matchers, where (1) indicates the terminological algorithm plus WordNet, (2) indicates the domain matcher, (3) is the learning matcher, and (4) indicates the structural matcher. For example, in the 'ID' case and the terminological matcher, the structural matcher generates 17 new suggestions of which no suggestion is correct, one suggestion is wrong, and 16 are inferred suggestions. In this evaluation the structural

Table 4. Structural matcher

Case	(1)+(4)	(2)+(4)	(3)+(4)
B	0/0/0/0	0/0/0/0	0/0/0/0
ID	17/0/1/16	17/0/1/16	0/0/0/0
nose	0/0/0/0	0/0/0/0	0/0/0/0
ear	19/0/2/17	21/0/2/19	2/0/0/2
eye	16/0/0/16	16/0/0/16	20/0/0/16

Table 5. Combination of algorithms

Case	E.Sugs	(1)+(2)	(1)+(3)	(2)+(3)	(1)+(2)+(3)
B	4	4/4/0	4/4/0	4/4/0	4/4/0
ID	8	4/4/0	6/6/0	5/5/0	6/6/0
nose	7	7/7/0	7/7/0	7/7/0	7/7/0
ear	29	28/28/0	28/28/0	28/28/0	29/29/0
eye	30	22/22/0	21/21/0	21/21/0	22/22/0

matcher is actually not needed in the 'B' case as the other algorithms already performed perfectly. In the other cases the structural matcher only returned inferred suggestions and some wrong suggestions. The fact that no more correct suggestions are found may be explained by the fact that the missing suggestions concern concepts in completely different positions in the two hierarchies. For other missing suggestions the concepts have a common ancestor or common descendants, but the ancestor or descendants are too distant for the similarity values to be influenced.

Table 5 presents the quality of the suggestions considering the combination of the different matchers. In table 5 we do not include the structural matcher because of its poor quality. In the evaluation we observe that if we carefully assign the weights for the matchers, the combination can always eliminate the wrong suggestions but still keep all the correct suggestions generated by the respective matchers. In the 'ID' case the combination also contributes a new correct suggestion. For instance, when combining the three matchers we obtain the best results in our experiments for all cases when we assign the weights 1.2, 2.0 and 1.6. Lower weights lead to the loss of correct suggestions and higher weights generate a number of wrong suggestions.

An advantage of using a system like SAMBO is that one can experiment with different (combinations of) strategies and different (combinations of) types of ontologies. For instance, our evaluation gives an indication about what (combinations of) strategies may work well for aligning ontologies with similar properties as our test ontologies. For instance, in our tests the terminological matcher gives good results while the best results are obtained by combining all (1, 2 and 3) matchers. However, when choosing a strategy other factors may also play a role. For instance, the combination strategy is more time consuming than the strategy using only the terminological matcher.

6   Conclusions

In this paper we have shown that different kinds of strategies are used by current alignment systems. We presented a framework for aligning ontologies where these different strategies can be combined. The framework can be used as a basis for building ontology alignment systems. We exemplified this by describing SAMBO, a system that is developed according to the framework and that implements different strategies.

Further, the framework can be used to experiment with combinations of strategies. This is a first step towards a general framework that can be used for comparative evaluations of alignment strategies. In this paper we experimented with different strategies and their combinations and showed results for well-known bio-ontologies.

In the future we will use the framework as a basis for implementing new strategies and test these strategies and their combinations using different types of ontologies. This will result in recommendations on which (combinations of) strategies are well suited for aligning which kinds of ontologies.

Acknowledgements

References

1. Collins F, Green E, Guttmacher A, Guyer M (2003) A vision for the future of genomics research. *Nature* 422:835-847.
2. Doan A, Madhavan J, Domingos P, Halevy A (2003) Ontology matching: A machine learning approach. Staab, Studer (eds) *Handbook on Ontologies in Information Systems*, pp 397-416, Springer.
3. Ehrig M, Staab S (2004) QOM - quick ontology mapping. *Proceedings of 3rd International Semantic Web Conference*, LNCS 3298, pp 683-697.
4. FaCT. http://www.cs.man.ac.uk/∼horrocks/FaCT/
5. The Gene Ontology Consortium (2000) Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25-29. http://www.geneontology.org/.
6. Gómez-Pérez A (1999) Ontological Engineering: A state of the Art. *Expert Update* 2(3):33-43.
7. Giunchiglia. F, Shvaiko P, Yatskevich M (2004) S-Match: an algorithm and an implementation of semantic matching. *Proceedings of the European Semantic Web Symposium*, LNCS 3053, pp 61-75.
8. KnowledgeWeb Consortium (2004) Deliverable 2.2.4 (Specification of a methodology, general criteria, and benchmark suites for benchmarking ontology tools). http://knowledgeweb.semanticweb.org/

9. Kotis K, Vouros GA (2004) The HCONE Approach to Ontology Merging. *The Semantic Web: Research and Applications, First European Semantic Web Symposium*, LNCS 3053, pp 137 - 151.

10. Lambrix P (2004) Ontologies in Bioinformatics and Systems Biology. Chapter 8 in Dubitzky W, Azuaje F (eds.) *Artificial Intelligence Methods and Tools for Systems Biology*, pp 129-146, Springer. ISBN: 1-4020-2859-8.

11. Le BT, Kuntz RD, Gandon F (2004) On ontology matching problem (for building a corporate semantic web in a multi-communities organization). *Proceedings of 6th International Conference on Enterprise Information Systems*.

12. Lambrix P, Edberg A (2003) Evaluation of ontology merging tools in bioinformatics. *Proceedings of the Pacific Symposium on Biocomputing*, pp 589-600.

13. Lambrix P, Tan H (2005) Merging DAML+OIL Ontologies. Barzdins, Caplinskas (eds) *Databases and Information Systems - Selected Papers from the Sixth International Baltic Conference on Databases and Information Systems*, pp 249-258, IOS Press.

14. McGuinness D, Fikes R, Rice J, Wilder S (2000) An Environment for Merging and Testing Large Ontologies. *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning*, pp 483-493.

15. Mitra P, Wiederhold G (2002) Resolving terminological heterogeneity in ontologies. *Proceedings of the ECAI Workshop on Ontologies and Semantic Interoperability*.

16. Noy NF, Musen M (2000) PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. *Proceedings of Seventeenth National Conference on Artificial Intelligence*, pp 450-455.

17. Noy NF, Musen M (2001) Anchor-PROMPT: Using Non-Local Context for Semantic Matching. *Proceedings of the IJCAI Workshop on Ontologies and Information Sharing*, pp 63-70.

18. Noy NF, Musen M (2002) Evaluating Ontology-Mapping Tools: Requirements and Experience. *Proceedings of the EKAW Workshop on Evaluation of Ontology Tools*.

19. Open Biomedical Ontologies. http://obo.sourceforge.net/

20. OntoWeb Consortium (2002) Deliverables 1.3 (A survey on ontology tools) and 1.4 (A survey on methodologies for developing, maintaining, evaluating and reengineering ontologies). http://www.ontoweb.org

21. PubMed. http://www.ncbi.nlm.nih.gov/entrez/query.fcgi

22. Pinto HS, Gómez-Pérez A, Martins JP (1999) Some Issues on Ontology Integration. *Proceedings of the IJCAI Workshop on Ontologies and Problem-Solving Methods*.

23. REWERSE. Backofen R, Badea M, Burger A, Fages F, Lambrix P, Nutt W, Schroeder M, Soliman S, Will S (2004) State-of-the-art in Bioinformatics. REWERSE Deliverable A2-D1.

24. REWERSE. Backofen R, Badea M, Barahona P, Burger A, Dawelbait G, Doms A, Fages F, Hotaran A, Jakonienė V, Krippahl L, Lambrix P, McLeod K, Möller S, Nutt W, Olsson B, Schroeder M, Soliman S, Tan H, Tilivea D, Will S (2005) Usage of bioinformatics tools and identification of information sources. REWERSE Deliverable A2-D2.

25. Racer. http://www.cs.concordia.ca/∼haarslev/racer/

26. Stevens R, Goble C, Bechhofer S (2000) Ontology-based knowledge representation for bioinformatics. *Briefings in Bioinformatics* 1(4):398-414.

27. Su XM, Hakkarainen S, Brasethvik T (2004) Semantic enrichment for improving systems interoperability. *Proceedings of the 2004 ACM Symposium on Applied Computing*, pp 1634-1641.

28. Stumme G, Mädche A (2001) FCA-Merge: Bottom-up merging of ontologies. *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pp 225-230.
29. Sushama P, Yun P, Timothy F (2002) Using Explicit Information To Map Between Two Ontologies, *Proceedings of the AAMAS 2002 Workshop on Ontologies in Agent Systems.*
30. Takai-Igarashi T, Nadaoka Y, Kaminuma T (1998) A Database for Cell Signaling Networks. *Journal of Computational Biology* 5(4):747-754.
31. UMLS. http://www.nlm.nih.gov/research/umls/about_umls.html
32. WordNet. http://wordnet.princeton.edu/
33. Kalfoglou Y, Schorlemmer M (2003) IF-Map: an ontology mapping method based on information flow theory. *Journal of data semantics*, 1:98-127.