Alignment of Biomedical Ontologies Using Life Science Literature

He Tan, Vaida Jakonienė, Patrick Lambrix,
Johan Aberg, and Nahid Shahmehri

Department of Computer and Information Science,
Linköpings universitet, SE-581 83 Linköping, Sweden

Abstract. In recent years many biomedical ontologies have been developed and many of these ontologies contain overlapping information. To be able to use multiple ontologies they have to be aligned. In this paper we propose strategies for aligning ontologies based on life science literature. We propose a basic algorithm as well as extensions that take the structure of the ontologies into account. We evaluate the strategies and compare them with strategies implemented in the alignment system SAMBO. We also evaluate the combination of the proposed strategies and the SAMBO strategies.

1 Introduction

Ontologies (e.g. [Lam04, Gom99]) 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 the work on biomedical ontologies is recognized as essential in some of the grand challenges of genomics research [CGGG03] and there is much international research cooperation for the development of ontologies (e.g. the Gene Ontology (GO) [GO00] and Open Biomedical Ontologies (OBO) [OBO] efforts) and the use of ontologies for the Semantic Web (e.g. the EU Network of Excellence REWERSE [REWERSE], [Lam05]).

Many ontologies have already been developed and many of these ontologies contain overlapping information. Often we would therefore want to be able to

E.G. Bremer et al. (Eds.): KDLL 2006, LNBI 3886, pp. 1–17, 2006.

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 (concepts and relations) in the different ontologies. These relationships can also be used in information integration [JL05]. It has been realized that ontology alignment, i.e. finding relationships between terms in the different ontologies, is a major issue and some organizations (e.g. the organization for Standards and Ontologies for Functional Genomics (SOFG)) have started to deal with it.

In this paper we present instance-based strategies for aligning biomedical ontologies. We focus on equivalence and is-a relationships. In section 3 we present an algorithm based on naive Bayes classifiers as well as extensions that take the structure of the ontologies into account. The strategies use life science literature and build on the intuition that a similarity measure between concepts can be computed based on the probability that documents about one concept are also about the other concept. Section 4 describes different experiments regarding the quality and performance of the proposed strategies and the combination of these strategies with other existing strategies. We describe related work in section 5 and conclude the paper in section 6. In the next section we provide background information on biomedical ontologies and ontology alignment systems.

2 Background

2.1 Biomedical Ontologies

In recent years many biomedical ontologies have been developed and the field has matured enough to develop standardization efforts. An example of this is the organization of the first SOFG conference 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 biomedical ontologies are available via OBO and there are many overlapping ontologies 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-data sources 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 [TNK98] 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, cataloging, 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 to generate an anatomy ontology (controlled vocabulary) covering the lifespan of the laboratory mouse. It organizes anatomical structures spatially and functionally, using is-a and part-of relationships.

2.2 Ontology Alignment 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 merge systems, i.e. they can create a new ontology based on the source ontologies and the alignment relationships. Many ontology alignment systems can be described as instantiations of the general framework defined in [LT05b] (figure 1). An alignment algorithm receives as input two source ontologies. The algorithm can include several matchers. These matchers calculate similarities between the terms from the different source ontologies. The matchers can implement strategies based on linguistic matching, structure-based strategies, constraint-based approaches, instance-based strategies, strategies that use auxiliary information or a combination of these.

Alignment suggestions are then determined by combining and filtering the results generated by one or more matchers. The pairs of terms 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 then presented to the user who accepts or rejects them. The acceptance and rejection of a suggestion may influence further suggestions. Further, a conflict checker is used to avoid conflicts introduced by the alignment relationships. The output of the alignment algorithm is a set of alignment relationships between terms from the source ontologies.

To date comparative evaluations of ontology alignment and merge systems have been performed by few groups ([OntoWeb] and [LE03, LT05a, LT05b, LT06]) and only the latter has focused on the quality of the alignment. Further, an ontology alignment contest was held at EON-2004 [Euz04]. The main goal of the contest was to show how ontology alignment tools can be evaluated and a follow-up was planned. An overview of alignment systems and a comparison between different alignment strategies can be found in [LT05a, LT05b, LT06].

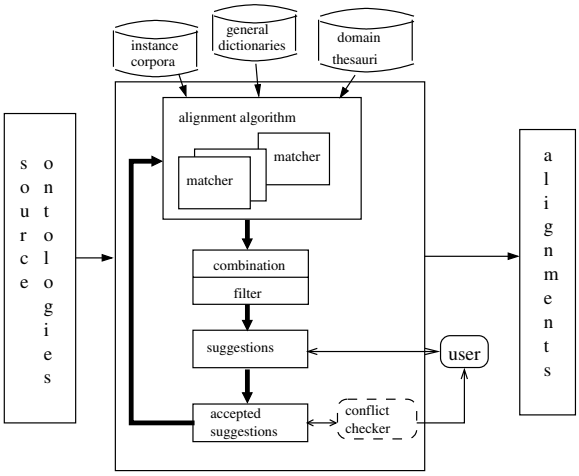

Fig. 1. A general alignment strategy [LT05b]

2.3 SAMBO

SAMBO[1] (System for Aligning and Merging Biomedical Ontologies) is an alignment and merge system for biomedical ontologies developed using the general framework defined in [LT05b]. The current implementation supports ontologies in OWL and DAML+OIL formats. Several kinds of matchers are used [LT05a, LT05b, LT06], including the basic algorithm described in this paper. These matchers can be combined using different weights and a threshold for filtering can be set. For each alignment suggestion the user can decide whether the terms are equivalent, there is an is-a relation between the terms, or the suggestion is rejected (figure 2). At each point in time during the alignment process 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 still need to be processed. In addition to the suggestion mode, the system has a manual mode in which the user can view the ontologies and manually align terms.

3 Alignment Algorithms

In this paper we present algorithms for suggesting alignments between two biomedical ontologies which focus on relationships between concepts. The algorithms make use of life science literature that is related to these concepts. They build on the intuition that a similarity measure between concepts in different ontologies can be computed based on the probability that documents about one concept are also about the other concept and vice versa. We propose a basic algorithm as well as two extensions that take the structure of the ontologies into account.

[1] The home page for SAMBO is http://www.ida.liu.se/∼iislab/projects/SAMBO/

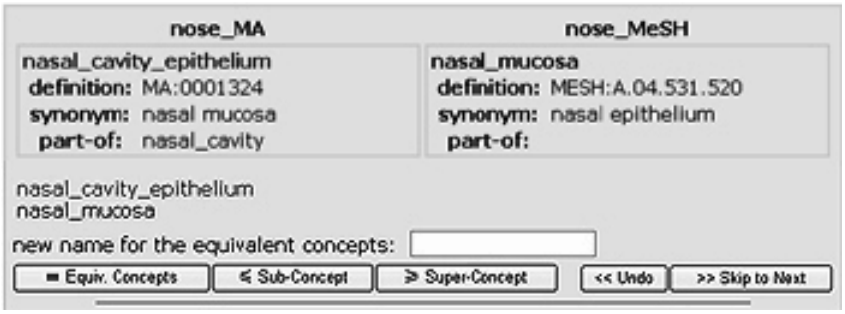

Fig. 2. Alignment suggestion

The algorithms contain the following basic steps.

1. **Generate corpora.** For each ontology that we want to align we generate a corpus of PubMed abstracts. PubMed [PubMed] is a service of the National Library of Medicine that includes over 15 millions citations from MEDLINE [MEDLINE] and other biomedical journals.
2. **Generating the classifiers.** For each ontology a document classifier is generated. This classifier returns for a given document the concept that is most closely related. To generate a classifier the corpus of abstracts associated to the classifier's ontology is used.
3. **Classification.** Documents of one ontology are classified by the document classifier of the other ontology and visa versa.
4. **Calculate similarities.** A similarity measure between concepts in the different ontologies is computed.

In the steps 2 and 3 of our algorithms we use (variants of) the naive Bayes classification algorithm. In the remainder of this section we describe the intuition behind a naive Bayes classifier for classifying text documents with respect to ontological concepts and present the different algorithms in more detail.

3.1 Text Classification Based on Naive Bayes Classifier

A naive Bayes classifier classifies a document d as related to a concept C in an ontology if the highest value for the posterior probability of a concept given the document d is obtained for the concept C. The posterior probability of concept C given document d is estimated using Bayes' rule [Mit97]:

$$P(C|d) = \frac{P(C)P(d|C)}{P(d)}$$

As $P(d)$ is independent of the concepts, it can be ignored. Also, the logarithm of the probability is often computed instead of the actual probability. This gives:

$$logP(C|d) \approx logP(C)P(d|C) = logP(C) + logP(d|C)$$

To evaluate the probabilities, the previously learned knowledge about the training documents originally associated to the ontological concepts is used. $P(C)$ is estimated by the ratio of the number of documents originally associated with C ($n_D(C)$) and the total number of documents related to the concepts in the ontology.

$$P(C) = \frac{n_D(C) + \lambda}{\sum_i n_D(C_i) + \lambda|O|}$$

where $0 < \lambda \leq 1$ is the Laplace smoothing parameter[2], and $|O|$ is the total number of concepts in the ontology. The term $P(d|C)$ is estimated by the probability that words w in the document d occur in documents originally related to C. Assuming that word occurrences are independent of occurrences of other words, we have

$$P(d|C) = \prod_{w \in d} P(w|C)$$

Let $n_W(C, w)$ be the number of occurrences of word w in documents associated with C, and $n_W(C) = \sum_w n_W(C, w)$ be the total number of occurrences of words in documents associated with C. Then $P(w|C)$ is estimated by

$$P(w|C) = \frac{n_W(C, w) + \lambda}{n_W(C) + \lambda|V|}$$

where λ is the earlier defined Laplace smoothing parameter, and $|V|$ is the size of the vocabulary, i.e. the number of distinct words in all of the training documents.

3.2 Basic Algorithm

We now describe the different steps in the basic algorithm in more detail.

1. **Generate corpora.** For each ontology we generate a corpus based on documents that are related to the concepts in the ontology. For each concept we use the concept name as a query term for PubMed and retrieve abstracts of documents that contain the query term in their title or abstract using the programming utilities [SW] provided by the retrieval system Entrez [Entrez]. A maximum number of retrieved abstracts per concept needs to be set beforehand.
2. **Generating the classifiers.** For each ontology a naive Bayes classifier[3] is generated. During the classifier generation $P(C)$ and $P(w|C)$ are calculated for every concept based on the corpus of abstracts associated to the ontology.
3. **Classification.** The naive Bayes classifier for one ontology is applied to every abstract in the abstract corpus of the other ontology and vice versa. For every abstract the classifier calculates $logP(C|d)$ with respect to every concept and classifies the abstract to the concept with the highest value for the posterior probability. The classifier keeps track of how the abstracts associated to concepts in one ontology are distributed over concepts in the other ontology.

[2] In the implementation $\lambda = 1$.

[3] The implementation of the naive Bayes classifier is based on the code available at http://www.cs.utexas.edu/users/mooney/ir-course/

4. **Calculate similarities.** As the last step we compute a similarity between concepts in different ontologies. We define the similarity between a concept C_1 from the first ontology and a concept C_2 from the second ontology as:

$$sim(C_1, C_2) = \frac{n_{NBC2}(C_1, C_2) + n_{NBC1}(C_2, C_1)}{n_D(C_1) + n_D(C_2)}$$

where $n_D(C)$ is the number of abstracts originally associated with C, and $n_{NBCx}(C_p, C_q)$ is the number of abstracts associated with C_p that are also related to C_q as found by classifier $NBCx$ related to ontology x.

The pairs of concepts with a similarity measure greater or equal than a predefined threshold are then presented to the user as candidate alignments.

3.3 Structure-Based Extensions of the Basic Algorithm

Most biomedical ontologies are organized using is-a relations. Therefore, the ontologies have inherent information about concepts and sub-concepts and this information could be used during the alignment process. In this section we propose extensions of the basic algorithm that take the structure (is-a relations) of the original ontologies into account by assuming that abstracts that are related to the sub-concepts of a concept C are also related to concept C. The first extension takes the structure into account during the generation of the classifiers. In the second extension we use a different similarity measure.

Structure-based classifier. This algorithm extends the classifier generation step (step 2) of the basic algorithm. To calculate the posterior probability of concept C given document d while taking into account the structure of the classifier's ontology, $P(C)$ and $P(w|C)$ are defined as follows (with $n_D(C)$, $|O|$, λ, $n_W(C, w)$ and $n_W(C)$ as defined before).

$$P(C) = \frac{\sum_{C_j \subseteq C} n_D(C_j) + \lambda}{\sum_i n_D(C_i) + \lambda|O|}, P(w|C) = \frac{\sum_{C_i \subseteq C} n_W(C_i, w) + \lambda}{\sum_{C_i \subseteq C} n_W(C_i) + \lambda|V|}$$

This is equivalent to extending the corpus of documents for each concept C in the ontology by including the documents related to the sub-concepts of the concept C, and then calculating the posterior probabilities using the previously defined naive Bayes classifier.

Structure-based similarity measure. A structure-based similarity between a concept C_1 from the first ontology and a concept C_2 from the second ontology can be defined as (with $n_D(C)$ and $n_{NBCx}(C_p, C_q)$ as defined before):

$$sim_{struct}(C_1, C_2) = \frac{\sum_{C_i \subseteq C_1, C_j \subseteq C_2} n_{NBC2}(C_i, C_j) + \sum_{C_i \subseteq C_1, C_j \subseteq C_2} n_{NBC1}(C_j, C_i)}{\sum_{C_i \subseteq C_1} n_D(C_i) + \sum_{C_j \subseteq C_2} n_D(C_j)}$$

In this definition the similarity between concepts is computed based on the naive Bayes classifiers applied to the concepts and their sub-concepts.

4 Evaluation

In our evaluation we have focused on several aspects. First, we investigated the influence of the number of PubMed abstracts on the quality of the suggestions. Further, we compare the proposed algorithms with respect to the quality of the suggestions they generate and the time they take to generate the suggestions. We also compare them to other matchers implemented in SAMBO with respect to the quality of the suggestions and investigate the combination of the proposed algorithms and the other SAMBO matchers.

4.1 Set-Up

Test cases. In the evaluation we create five cases from several well-known biomedical ontologies. For the first two cases we use a part of a GO ontology [GO00] together with a part of SigO [TNK98]. Each case was chosen in such a way that there is an overlap between the GO part and the SigO part. The first case, *B* (behavior), contains 57 terms from GO and 10 terms from SigO. The second case, *ID* (immune defense), contains 73 terms from GO and 17 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 biomedical ontologies that are available from OBO [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 relations. The synonyms were transformed into equivalence statements. Domain experts were asked to analyze the cases and provide alignment relationships based on equivalence and is-a relations. In our evaluations we have used the ontologies and the alignment relationships from the experts as they were provided to us.

Table 1. Number of abstracts

Ontology	concepts	100	90-99	80-89	70-79	60-69	50-59	40-49	30-39	20-29	10-19	1-9	0
B-GO	57	37	0	2	2	2	0	0	0	1	3	4	6
B-SigO	10	10	0	0	0	0	0	0	0	0	0	0	0
ID-GO	73	44	1	2	0	3	0	1	0	3	4	1	14
ID-SigO	17	17	0	0	0	0	0	0	0	0	0	0	0
nose-MA	18	13	0	0	0	1	0	1	0	0	0	3	0
nose-MeSH	15	15	0	0	0	0	0	0	0	0	0	0	0
ear-MA	77	48	1	0	1	3	0	1	2	2	5	9	5
ear-MeSH	39	34	0	1	0	0	0	0	0	1	1	1	1
eye-MA	112	84	2	1	1	3	3	0	1	2	7	8	0
eye-MeSH	45	37	0	0	0	0	2	0	1	1	1	1	2

PubMed. For the generation of the corpora we used PubMed as it was available on October 23, 2005. (All corpora generated between 11.42 and 13.15 CEST.) We generated different corpora for each ontology by assigning a maximum of 20, 40, 60, 80 and 100 PubMed abstracts, respectively, for each concept in the source ontologies. The corpus generated using a maximum of 20 abstracts is a sub-set of the corpus generated using a maximum of 40 abstracts, and similarly for 60, 80 and 100. The retrieval system for PubMed did not always find the allowed number of abstracts for all concepts. Table 1 shows for how many concepts the system retrieved 100, between 90 and 99, ... , between 1-9 and no abstracts. When more abstracts than allowed were found, we retrieved the most recent abstracts, otherwise all abstracts were retrieved. We observe that when the allowed number of abstracts is 100, in some cases only for 60% of the concepts 100 abstracts were retrieved. Even when 20 abstracts were allowed, not for all concepts this number of abstracts was retrieved. In this experiment there is no apparent relationship between the location of a concept in the is-a hierarchy and how many abstracts are retrieved for that concept.

4.2 Evaluation Results

Influence of the number of PubMed abstracts on the quality of the suggestions. In table 2 we present the quality of suggestions generated by the basic algorithm with different numbers of abstracts. The cases are given in the first column. The second column represents the number of expected suggestions provided by domain experts. In our evaluation we consider only expected suggestions related to equivalence of terms or is-a relations between terms. For instance, in the ear case, there are 27 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 are counted 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 equivalent to 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 third column represents the threshold value. Pairs with a similarity value higher than the threshold are suggestions. The other columns present the results. The four numbers in the cells represent the number of suggestions provided by the algorithm, the number of correct suggestions, the number of wrong suggestions and the number of inferred suggestions, respectively. For instance, for the case B with a maximum of 100 PubMed abstracts per concept and threshold 0.4 the algorithm generates 4 suggestions of which 2 suggestions are correct, 1 suggestion is wrong and 1 suggestion is inferred.

For the threshold 0.4 the precision[4] usually becomes higher when the maximum number of abstracts increases, e.g. in the case eye the precision goes up

[4] We use precision as it is usually defined in information retrieval, i.e. the number of correct suggestions divided by the number of suggestions. As noted before, inferred suggestions are counted neither correct nor wrong. Similarly, recall is defined as the number of correct suggestions divided by the total number of correct suggestions, in this case the expected suggestions.

Table 2. Influence of number of abstracts. (The cells a/b/c/d represent the number of a) suggestions, b) correct suggestions, c) wrong suggestions and d) inferred suggestions).

Case	ES	*Th*	20	40	60	80	100
B	4	0.4	3/2/0/1	3/2/1/0	6/2/1/3	4/2/0/2	4/2/1/1
		0.5	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0
		0.6	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0
		0.7	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0
		0.8	2/2/0/0	2/2/0/0	2/2/0/0	2/2/0/0	1/1/0/0
ID	8	0.4	11/4/4/3	9/5/3/1	10/6/3/1	9/6/3/0	9/6/3/0
		0.5	7/4/0/3	6/5/0/1	7/5/1/1	5/5/0/0	5/5/0/0
		0.6	5/4/0/1	4/3/0/1	2/2/0/0	2/2/0/0	2/2/0/0
		0.7	2/2/0/0	1/1/0/0	1/1/0/0	1/1/0/0	1/1/0/0
		0.8	0/0/0/0	0/0/0/0	0/0/0/0	0/0/0/0	0/0/0/0
nose	7	0.4	7/5/2/0	6/5/1/0	6/5/1/0	6/5/1/0	6/5/1/0
		0.5	6/5/1/0	5/5/0/0	6/5/1/0	6/5/1/0	6/5/1/0
		0.6	5/5/0/0	5/5/0/0	5/5/0/0	5/5/0/0	5/5/0/0
		0.7	5/5/0/0	5/5/0/0	5/5/0/0	5/5/0/0	5/5/0/0
		0.8	4/4/0/0	5/5/0/0	3/3/0/0	3/3/0/0	3/3/0/0
ear	27	0.4	20/16/4/0	19/16/3/0	19/16/3/0	18/16/2/0	18/16/2/0
		0.5	18/16/2/0	17/15/2/0	15/14/1/0	15/14/1/0	15/14/1/0
		0.6	14/14/0/0	15/14/1/0	11/10/1/0	12/11/1/0	12/11/1/0
		0.7	11/11/0/0	11/10/1/0	11/10/1/0	11/10/1/0	11/10/1/0
		0.8	5/5/0/0	5/5/0/0	4/4/0/0	3/3/0/0	3/3/0/0
eye	27	0.4	33/19/13/1	32/18/13/1	27/18/9/0	27/19/8/0	25/18/7/0
		0.5	20/17/3/0	20/18/2/0	20/17/3/0	18/16/2/0	18/17/1/0
		0.6	16/16/0/0	17/16/1/0	15/14/1/0	14/14/0/0	14/14/0/0
		0.7	12/12/0/0	11/11/0/0	13/13/0/0	11/11/0/0	10/10/0/0
		0.8	5/5/0/0	5/5/0/0	5/5/0/0	5/5/0/0	3/3/0/0

from 0.575 to 0.72. The exception is the case B where we obtain the best result when the maximum number of abstracts is 20 or 40. As the maximum number of abstracts increases, no more correct suggestions are found, except in the ID case where two more correct *is-a* relationships are found when the maximum number of abstracts is higher than 40. For higher threshold values the number of suggestions diminishes, e.g. in the ID case where the number of suggestions goes down to 0 and both correct and wrong suggestions are filtered out. When the maximum number of abstracts increases, the number of correct suggestions goes down even faster when the threshold becomes higher. The results of this experiment suggest that the quality of the suggestions does not necessarily become better when we have larger corpora of abstracts. The experiment also shows that the corpora have an impact on the quality of the suggestions.

Quality of the suggestions. In table 3 we compare the quality of the suggestions generated by our basic (Basic) algorithm, the extension that takes the structure into account during the generation of the classifier (StrucCl), the ex-

Table 3. Comparison of matchers: quality of the suggestions

Case	ES	*Th*	Basic	StrucCl	StrucSim	StrucClSim
B	4	0.4	4/2/1/1	5/2/0/3	20/3/6/11	7/1/0/6
		0.5	2/2/0/0	3/2/0/1	7/1/2/4	5/1/0/4
		0.6	2/2/0/0	2/2/0/0	5/1/0/4	4/0/0/4
		0.7	2/2/0/0	1/1/0/0	3/1/0/2	2/0/0/2
		0.8	1/1/0/0	0/0/0/0	2/0/0/2	1/0/0/1
ID	8	0.4	9/6/3/0	4/3/0/1	14/6/4/4	5/2/1/2
		0.5	5/5/0/0	2/2/0/0	9/5/2/2	4/1/1/2
		0.6	2/2/0/0	0/0/0/0	5/2/1/2	4/1/1/2
		0.7	1/1/0/0	0/0/0/0	4/1/1/2	4/1/1/2
		0.8	0/0/0/0	0/0/0/0	4/1/1/2	3/0/1/2
nose	7	0.4	6/5/1/0	7/5/2/0	9/5/2/2	8/5/2/1
		0.5	6/5/1/0	5/5/0/0	6/4/1/1	4/3/0/1
		0.6	5/5/0/0	5/5/0/0	4/3/0/1	3/3/0/0
		0.7	5/5/0/0	2/2/0/0	3/3/0/0	1/1/0/0
		0.8	3/3/0/0	2/2/0/0	1/1/0/0	1/1/0/0
ear	27	0.4	18/16/2/0	15/11/2/2	24/12/4/8	14/8/3/3
		0.5	15/14/1/0	6/5/0/1	16/11/1/4	5/4/0/1
		0.6	12/11/1/0	3/3/0/0	12/10/1/1	1/1/0/0
		0.7	11/10/1/0	1/1/0/0	10/9/1/0	0/0/0/0
		0.8	3/3/0/0	1/1/0/0	2/2/0/0	0/0/0/0
eye	27	0.4	25/18/7/0	25/11/11/3	34/14/15/5	16/8/6/2
		0.5	18/17/1/0	8/7/1/0	21/12/6/3	10/4/4/2
		0.6	14/14/0/0	3/3/0/0	14/9/4/1	1/0/1/0
		0.7	10/10/0/0	1/1/0/0	9/7/2/0	1/0/1/0
		0.8	3/3/0/0	1/1/0/0	3/3/0/0	0/0/0/0

tension that uses the structure-based similarity measure (StrucSim) and an algorithm using both extensions (StrucClSim). In the evaluation we generated corpora for each ontology by assigning a maximum of 100 PubMed abstracts for each concept in the source ontologies.

In most of the cases Basic outperforms the structure-based algorithms. Only in very few cases the structure-based algorithms showed higher precision and recall. For the B, ID, and some settings of the nose, ear and eye cases, StrucSim returns the largest number of suggestions. A large part of these suggestions are inferred or wrong. The lowest number of suggestions is returned by StrucCl and StrucClSim. Among the structure-based algorithms, StrucSim generates the largest number of correct suggestions for the ID, ear and eye cases. In some cases of B and nose, StrucSim is outperformed by StrucCl. The only new correct suggestion is found by StrucSim and StrClSim for the ear case with the threshold 0.5 - (auditory bone, ear ossicle). The concepts in the suggestion have common sub-concepts. For the structure-based algorithms we noted that the similarity value for a pair of concepts depends to a large degree on the content of the abstracts related to the sub-concepts of the concepts in the suggestion (StrucCl, StrucClSim) and on how the abstracts of one ontology are classified

by the classifier of the other ontology (StrucSim, StrucClSim). This dependency is the source of improved results in some of the test cases, but it may also result in the decreased quality of the results.

The results of StrucCl illustrated that the use of the abstracts of sub-concepts may cause both positive and negative results. Some of the wrong suggestions were removed based on the fact that abstracts related to sub-concepts dealt with unrelated topics. For instance, (circadian rhythm, sleep response) in B gets a lower similarity value in StrucCl than in Basic. However, at the same time new wrong suggestions were introduced. For instance, (reproductive behavior, feeding behavior) is a wrong suggestion for B. In this case abstracts related to their sub-concepts are about behavior. Also, some of the correct suggestions were removed by the algorithm. In some of these cases the abstracts of the sub-concepts included many more non-relevant concepts or very little relevant concepts, which caused a decrease in probabilities for the important terms in the abstracts. Some of the correct suggestions were removed because a number of abstracts classified to certain concepts by Basic were classified to other concepts by StrucCl.

Many of the suggestions generated by StrucSim are inferred suggestions, which illustrates the fact that sub-concepts may give strong support for the analyzed concepts. However, they may also be the cause of wrong suggestions.

For instance, (defense response, body level function) is a wrong suggestion in the ID case and both concepts have immune response as their sub-concept. In our algorithms we did not propagate the abstracts via part-of. This caused several wrong suggestions in the MA-MeSH cases as part-of and is-a are used differently in the two ontologies.

StrucSimCl combines the StrucCl and StrucSim approaches. In none of the cases StrucSimCl returned higher similarity values for the correct suggestions than the other two structure-based algorithms. We observed that low similarity values of StrucCl or StrucSim have a large influence on the final similarity. In a number of cases StrucClSim returns low similarity values even though StrucCl and StrucSim return high similarities when executed separately. The poor performance of the algorithm could be explained by the fact that an abstract can be classified to only one concept. In some cases this results in abstracts previously classified to a concept to be classified to a super-concept.

Some expected suggestions were not found by any of our instance-based algorithms. One reason could be the low number of abstracts in the generated corpora for some concepts. For instance, in the eye case, (macula, macula lutea) is not returned, where macula lutea has only 26 related abstracts. Also, the abstracts in the corpora may cover different domains.

Performance of the algorithms. We also evaluated the time it takes for the discussed algorithms to compute the suggestions. For all of the algorithms we generated the PubMed corpora beforehand. The time for loading the ontologies ranges from 0.9 to 3.2 seconds. In most of the cases the time is around 1.5 seconds. In table 4 we present the time to generate suggestions. This covers the time for learning the classifier and the time for computing the similarity values. In the nose case, where a small number of abstracts is classified and the

Table 4. Comparison of matchers: time for computation of the suggestions (in seconds)

Case	Basic	StrucCl	StrucSim	StrucClSim
B	421.2	567.3	497.8	570.4
ID	354.1	824.2	603.0	1001.0
nose	216.1	219.1	213.1	223.3
ear	667.5	1097.1	904.5	1032.5
eye	1848.1	2012.5	1982.5	2024.8

ontologies contain only few is-a relations, there is no significant difference in the performance of the algorithms. For larger cases there is a tendency that there is an increase of time from Basic to StrucSim to StrucCl to StrucClSim. We used a SUN Ultra 5_10 Sparc workstation for these tests.

Quality of the suggestions compared to other algorithms. In table 5 we show the quality of the suggestions of other matchers that were presented in [LT06] (preliminary results in [LT05b]): a terminological matcher (Term), a terminological matcher using WordNet (TermWN), and a matcher (Dom) using

Table 5. Other matchers: quality of the suggestions [LT06]

Case	ES	*Th*	Term	TermWN	Dom
B	4	0.4	58/4/22/32	58/4/22/32	4/4/0/0
		0.5	35/4/13/18	35/4/13/18	4/4/0/0
		0.6	13/4/4/5	13/4/4/5	4/4/0/0
		0.7	6/4/0/2	6/4/0/2	4/4/0/0
		0.8	4/4/0/0	4/4/0/0	4/4/0/0
ID	8	0.4	96/7/66/23	96/7/66/23	4/4/0/0
		0.5	49/7/25/17	49/7/25/17	4/4/0/0
		0.6	15/5/4/6	16/5/5/6	4/4/0/0
		0.7	7/5/2/0	7/5/2/0	4/4/0/0
		0.8	6/4/0/2	6/4/0/2	4/4/0/0
nose	7	0.4	47/7/36/4	48/7/37/4	7/7/0/0
		0.5	27/7/17/3	28/7/18/3	7/7/0/0
		0.6	7/6/1/0	8/6/2/0	7/7/0/0
		0.7	6/6/0/0	6/6/0/0	6/6/0/0
		0.8	6/6/0/0	6/6/0/0	6/6/0/0
ear	27	0.4	147/26/104/17	155/26/110/19	26/23/2/1
		0.5	92/26/58/8	99/26/65/8	26/23/2/1
		0.6	47/26/19/2	47/26/19/2	26/23/2/1
		0.7	33/25/8/0	34/26/8/0	24/22/2/0
		0.8	26/24/2/0	28/25/3/0	24/22/2/0
eye	27	0.4	130/26/95/9	135/26/100/9	22/21/1/0
		0.5	72/23/42/7	74/23/44/7	22/21/1/0
		0.6	33/22/10/1	33/22/10/1	22/21/1/0
		0.7	24/21/3/0	24/21/3/0	19/18/1/0
		0.8	19/18/1/0	22/20/2/0	19/18/1/0

domain knowledge in the form of the Unified Medical Language System (UMLS) of the U.S. National Library of Medicine [UMLS]. The terminological matcher contains matching algorithms based on the names and synonyms of concepts and relations. The matcher is a combination matcher based on two approximate string matching algorithms (n-gram and edit distance) and a linguistic algorithm. In TermWN a general thesaurus, WordNet [WordNet], is used to enhance the similarity measure by using the hypernym relationships in WordNet. Dom uses the Metathesaurus in the UMLS which contains more than 100 biomedical and health-related vocabularies. The Metathesaurus is organized using concepts. The concepts may have synonyms which are the terms in the different vocabularies in the Metathesaurus that have the same intended meaning. The similarity of two terms in the source ontologies is determined by their relationship in UMLS. For more detailed information about these matchers we refer to [LT06].

We compare these matchers with Basic. The quality of the suggestions for Basic 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 Dom. However, in the eye case it produces the worst result. In this case all its correct suggestions are also found by the other matchers. We also note that the other matchers take synonyms into account and as our test ontologies contain many synonyms, their results improve considerably.

Combination with other matchers. Table 6 presents the quality of the suggestions considering the combination of the different matchers. The suggestions are determined based on the combination of the similarity values measured by individual matchers using weights, $sim(C_1, C_2) = (\sum_{k=1}^{n} w_k * sim_k(C_1, C_2)) / \sum_{k=1}^{n} w_k$, where sim_k and w_k represent the similarity values and weights, respectively, for the different matchers. In the experiment we used 1 as the weight for each matcher and 0.5 as the threshold value. The combination of our instance-based algorithms with Dom and TermWN leads to higher quality results. For the B, nose and ear cases, the instance-based algorithms combined with Dom return the same num-

Table 6. Combination of matchers

Case	ES	Matcher	Basic	StrucCl	StrucSim	StrucClSim
B	4	TermWN	6/4/0/2	5/4/0/1	10/4/2/4	6/4/0/2
		Dom	4/4/0/0	4/4/0/0	4/4/0/0	4/4/0/0
ID	8	TermWN	8/7/1/0	5/4/0/1	12/7/2/3	7/5/1/1
		Dom	4/4/0/0	4/4/0/0	4/4/0/0	4/4/0/0
nose	7	TermWN	8/7/1/0	8/7/1/0	10/7/2/1	8/7/1/0
		Dom	7/7/0/0	7/7/0/0	7/7/0/0	7/7/0/0
ear	27	TermWN	27/22/5/0	26/22/4/0	28/22/5/1	28/22/5/1
		Dom	24/22/2/0	24/22/2/0	24/22/2/0	24/22/2/0
eye	27	TermWN	24/21/3/0	23/19/4/0	30/21/8/1	21/19/2/0
		Dom	20/19/1/0	19/18/1/0	20/19/1/0	19/18/1/0

ber of correct suggestions as in the combination with TermWN. For the ID and eye cases the combination with TermWN gives better recall but lower precision. Dom tends to remove suggestions for which it finds no relationship in its domain knowledge. As could be expected from the results in table 3, StrucSim combined with TermWN returns more correct suggestions than StrucCl combined with TermWN at the expense of a larger number of wrong suggestions. All correct suggestions that are found by the combinations of matchers were also found by TermWN. The combinations of TermWN and Dom with the instance-based algorithms remove some of the wrong and inferred suggestions. In particular, for TermWN a large number of redundant suggestions were eliminated in the combination. However, at the same time some correct suggestions returned by TermWN and Dom were removed in the combination.

5 Related Work

Some ontology alignment and merging systems provide alignment strategies using literature, such as ArtGen [MW02], FCA-Merge [SM01] and OntoMapper [SYT02]. The basic alignment algorithm in ArtGen calculates the similarity between concepts based on their names which are seen as lists of words. One method to compute the similarity between a pair of words is based on the similarity between the contexts (1000-character neighborhoods) of all occurrences of the words in a set of domain-specific Web pages. In FCA-Merge the user constructs a merged ontology based on a concept lattice. The concept lattice is derived using formal concept analysis based on how documents from a given domain-specific corpus are classified to the concepts in the ontologies using natural language processing techniques. OntoMapper provides an ontology alignment algorithm using Bayesian learning. A set of documents (abstracts of technical papers taken from ACM's digital library and Citeseer) is assigned to each concept in the ontologies. Two raw similarity scores matrices for the ontologies are computed by the Rainbow text classifier. The similarity between the concepts is calculated based on these two matrices using the Bayesian method.

There are systems that implement alignment algorithms based on the structure of the ontologies. Most systems rely on the existence of previously aligned concepts. For instance, Anchor-PROMPT [NM01] determines the similarity of concepts by the frequency of their appearance along the paths between previously aligned concepts. The paths may be composed of any kind of relations. Also SAMBO as described in [LT05b] provides such a component where the similarity between concepts is augmented based on their location in the is-a hierarchy relative to already aligned concepts. However, for our test ontologies, these methods often did not perform well. In this paper we proposed methods that do not require previously aligned concepts. Also OntoMapper does not require previously aligned concepts and takes the documents from the sub-concepts into account when computing the similarity between two concepts. However, as this is hard-coded in the method, it is not clear how the structure of the ontologies influences the result of the computation.

6 Conclusion

In this paper we proposed and experimented with instance-based alignment strategies that use life science literature for aligning biomedical ontologies. We proposed a basic algorithm as well as extensions that take the structure of the ontologies into account. We evaluated the influence of the size of the literature corpus, the quality and performance of the strategies and their combination with other strategies. The basic algorithm outperforms the structure-based strategies in most cases, although compared to the other matchers in SAMBO the quality of the suggestions varies for the different cases. In some cases it produces the best result, in some cases the worst. An advantage of our structure-based strategies is that they can be used without information about previously aligned concepts, as many other systems require. However, the best results are obtained when the instance-based strategies are combined with other strategies. The other strategies usually provide new correct suggestions while the instance-based algorithms usually have the effect of removing wrong suggestions.

There are a number of issues that we still want to investigate. A limitation of our algorithms is that abstracts are only classified to one concept. We want to extend our strategies by allowing abstracts to be classified to 0, 1 or more concepts. We are also interested in looking at other classification algorithms. Regarding the structure the ontologies in the current experiments are reasonably simple taxonomies. We want to investigate whether the structure-based strategies lead to similar results for other types of ontologies. Further, our matchers could be enhanced to use synonyms and domain knowledge.

References

[CGGG03] Collins F, Green E, Guttmacher A, Guyer M (2003) A vision for the future of genomics research. *Nature*, 422: 835-847.

[Entrez] Entrez. http://www.ncbi.nlm.nih.gov/Database/index.html

[Euz04] Euzenat J (2004) Introduction to the EON ontology alignment context. *3rd Int. Workshop on the Evaluation of Ontology-based Tools.*

[GO00] The Gene Ontology Consortium (2000) Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25-29. http://www.geneontology.org/.

[Gom99] Gómez-Pérez A (1999) Ontological Engineering: A state of the Art. *Expert Update*, 2(3):33-43.

[JL05] Jakoniene V, Lambrix P (2005) Ontology-based Integration for Bioinformatics. *VLDB Workshop on Ontologies-based techniques for DataBases and Information Systems - ODBIS 2005*, pp 55-58.

[Lam04] 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.

[Lam05] Lambrix P (2005) Towards a Semantic Web for Bioinformatics using Ontology-based Annotation. *14th IEEE WET-ICE*, pp 3-7. Invited talk.

[LE03] Lambrix P, Edberg A (2003) Evaluation of ontology merging tools in bioinformatics. *Pacific Symposium on Biocomputing*, 8:589-600.

[LT05a] Lambrix P, Tan H, (2005) Merging DAML+OIL Ontologies. Barzdins, Caplinskas (eds) *Databases and Information Systems*, pp 249-258, IOS Press.

[LT05b] Lambrix P, Tan H (2005) A Framework for Aligning Ontologies. *3rd Workshop on Principles and Practice of Semantic Web Reasoning*, LNCS 3703, pp 17-31.

[LT06] Lambrix P, Tan H (2005) SAMBO - a System for Aligning and Merging Biomedical Ontologies. Submitted.

[MEDLINE] MEDLINE. http://www.nlm.nih.gov/databases/databases_medline.html

[Mit97] Mitchell T (1997) *Machine Learning*. McGraw-Hill.

[MW02] Mitra P, Wiederhold G (2002) Resolving terminological heterogeneity in ontologies. *ECAI Workshop on Ontologies and Semantic Interoperability*.

[NM01] Noy N, Musen M (2001) Anchor-PROMPT: Using Non-Local Context for Semantic Matching. *IJCAI Workshop on Ontologies and Information Sharing*, pp 63-70.

[OBO] Open Biomedical Ontologies. http://obo.sourceforge.net/

[OntoWeb] 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).

[Protege] Protégé. http://protege.stanford.edu/index.html

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

[REWERSE] REWERSE. http://www.rewerse.net

[SM01] Stumme G, Mädche A (2001) FCA-Merge: Bottom-up merging of ontologies. *17th IJCAI*, pp 225-230.

[SW] Sayers E, Wheeler D. Building Customized Data Pipelines Using the Entrez Programming Utilities (eUtils). *NCBI Coursework*.

[SYT02] Sushama S, Yun P, Timothy F (2002) Using Explicit Information To Map Between Two Ontologies. *AAMAS Workshop on Ontologies in Agent Systems*.

[TNK98] Takai-Igarashi T, Nadaoka Y, Kaminuma T (1998) A Database for Cell Signaling Networks. *Journal of Computational Biology* 5(4):747-754.

[UMLS] UMLS. http://www.nlm.nih.gov/research/umls/

[WordNet] WordNet. http://wordnet.princeton.edu/