

Mapping Composition for Matching Large Life Science Ontologies

Anika Groß^{1,2}, Michael Hartung^{1,2}, Toralf Kirsten^{2,3}, Erhard Rahm^{1,2}

¹Department of Computer Science, University of Leipzig, Germany

²Interdisciplinary Centre for Bioinformatics, University of Leipzig, Germany

³Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig, Germany

Abstract. There is an increasing need to interrelate different life science ontologies in order to facilitate data integration or semantic data analysis. Ontology matching aims at a largely automatic generation of mappings between ontologies mostly by calculating the linguistic and structural similarity of their concepts. In this paper we investigate an indirect computation of ontology mappings that composes and thus reuses previously determined ontology mappings that involve intermediate ontologies. The composition approach promises a fast computation of new mappings with reduced manual effort. Our evaluation for large anatomy ontologies shows that composing mappings via intermediate hub ontologies is not only highly efficient but can also achieve better match quality than with a direct matching of ontologies.

Keywords: ontology matching, matching quality, compose, reuse

1 Introduction

Ontologies have become increasingly important for life sciences, in particular to semantically annotate molecular-biological objects such as proteins or pathways [5, 16]. There are frequently multiple interrelated ontologies of a domain. For instance, information about mammalian anatomy can be found in Foundational Model of Anatomy (FMA) [8], NCI Thesaurus (NCIT) [19], or Adult Mouse Anatomy (MA) [1]. This situation led to a growing need to determine mappings between pairs of related ontologies. These *ontology mappings* are valuable for enhanced semantic data analysis, data integration [11], for merging (combining) the ontologies [14] and to support comparative science, e.g., mouse models for human cancer [18].

Since the manual creation of mappings is often too labor-intensive for large ontologies with thousands of concepts, matching approaches have been proposed to (semi-) automatically determine ontology mappings, e.g., by calculating the linguistic or structural similarity between ontology concepts [7]. Many ontology matching systems have been developed in recent years and several of them

participate in the annual Ontology Evaluation Alignment Initiative (OAEL) [21]. A promising direction to determine ontology mappings that has found only little attention so far (see Related Work) is the reuse of previously determined mappings that involve the ontologies to be matched. In particular, the composition of mappings with an intermediate ontology can be used to indirectly compute ontology mappings. For instance, an ontology mapping between MA and NCIT can be obtained by composing two existing mappings to an intermediate ontology, e.g., the Uber anatomy ontology (Uberon) [25] or Universal Medical Language System (UMLS) [26]. Fig. 1 exemplifies this approach for two selected concepts *MA_0001421* and *NCI_C32239* that are described by name and synonym attributes. A direct match of both concepts is non-trivial since their names differ significantly. However, using Uberon as an intermediate ontology allows us to reuse the correspondences (matches) *MA_0001421-UBERON:0001092* (exact match of ‘*atlas*’) and *UBERON:0001092-NCI_C32239* (exact match of ‘*C1 vertebra*’). The composition of these two correspondences results in the new correspondence between *MA_0001421* and *NCI_C32239*.

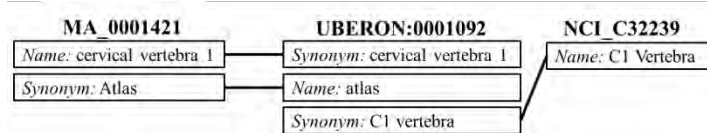


Figure 1. Composition of correspondences via an Uberon concept

Composing existing mappings for aligning two ontologies is promising in multiple ways. First, the life science community is continuously generating new mappings that are collected in repositories such as BioPortal [20] and can be reused for composition. Such a reuse is especially promising if the mappings are of high quality, e.g. validated by domain experts. Second, composing ontology mappings can likely be executed very fast while directly matching large ontologies is time-consuming and typically of quadratic complexity (every concept of the first is compared with every concept of the second ontology). Third, the intermediate ontology may contain additional knowledge which can be useful to detect further correspondences for improved mapping quality. Finally, if the intermediate ontology is a comprehensive ontology offering ontology mappings to numerous other ontologies we can consider it as a *hub ontology* providing large reuse potential. For a new ontology one only needs to create a mapping to the hub and use this for the composition to all other ontologies in the domain.

We therefore study composition-based matching of life science ontologies and make the following contributions:

- We propose a composition-based ontology matching approach which reuses previously determined mappings with one or multiple intermediate ontologies.
- Our approach is based on powerful ontology and mapping operators such as compose, match and extract. The approach also supports an incremental extension of composed mappings for improved match quality.
- We evaluate the approach by determining ontology mappings between the anatomy ontologies MA and NCIT utilizing large intermediate ontologies such as UMLS, FMA, Uberon and RadLex. The results demonstrate the high effectiveness and efficiency of composition-based ontology

matching.

In Sec. 2, we introduce our ontology/mapping model and discuss the ontology matching process. Sec. 3 defines the operators and shows their use within our composition-based match approach. We evaluate the approach in Sec. 4. After a discussion of related work (Sec. 5), we summarize and outline possible future work.

2 Preliminaries: Models and Ontology Matching

An ontology $O = (C, R)$ consists of a set of concepts C that are interconnected by relationships $r \in R$. Each ontology concept is described by a set of single- or multi-valued attributes. The concept name is the most common attribute in life science ontologies. Further common attributes include synonym (alternate name) and concept definition. A special attribute accession number c_{acc} is used to unambiguously identify a concept c within the ontology O . Relationships interconnecting concepts are of a certain kind such as *is_a* (e.g., ‘lung’ *is_a* ‘organ’) to represent specialization relationships or *part_of* (e.g., ‘left lung’ *part_of* ‘lung’) to represent part-whole relationships.

An ontology mapping $MO_1, O_2 = \{(c_1, c_2, sim) \mid c_1 \in O_1, c_2 \in O_2, sim \in [0, 1]\}$ between two ontologies O_1 and O_2 consists of a set of correspondences. Each correspondence (c_1, c_2, sim) interconnects two related or equivalent ontology concepts c_1 and c_2 . The strength of the connection is represented by a similarity value sim between 0 and 1. The greater the similarity value, the more similar are the corresponding concepts.

Ontology mappings can be created manually by domain experts. However, the complexity and size of the input ontologies make a manual creation often impossible. Thus, (semi-)automatic ontology matching approaches have been proposed [7]. They can roughly be classified into metadata- or schema-

based and instance-based approaches [23]. The metadata-based approaches are mostly either linguistic or structural matchers. Linguistic matchers typically employ string similarity measures, e.g., ExactMatch, n-gram or EditDistance, on the concept attributes (concept name, synonym, definition). Structural matchers also consider the ontology structure for matching, e.g., context information from children or ancestors of concepts. To improve the match quality compared to the adoption of a single matcher, current match systems such as COMA++ [3] or GOMMA [10, 13] support the flexible combination of multiple matchers and the aggregation of their results. In this paper we focus on linguistic matchers since previous works [9] has shown that they produce ontology mappings of good quality especially for anatomy ontologies that we consider here.

3 Mapping Composition

In this section, we present our composition-based match algorithm to indirectly match ontologies by reusing existing ontology mappings. We start with a discussion of the general idea of using intermediate ontologies in Sec. 3.1 and introduce our ontology and mapping operators in Sec. 3.2. We then combine the proposed operators in the composition-based match algorithm (Sec. 3.3).

3.1 Indirect Matching Via Intermediate Ontologies

The general idea of our approach is to use mappings to intermediate ontologies for indirect ontology matching. Such mappings

are typically produced in a resource-intensive match process, in particular, when the mappings or portions of them are created manually or computed by sophisticated match algorithms. Therefore, reusing such mappings promises to save or reduce the huge effort necessary when starting from scratch for matching two ontologies. Repositories such as BioPortal provide an increasing number of ontology mappings that can be used for a composition-based ontology matching.

Fig. 2a shows the basic situation consisting of two ontologies O_1 and O_2 as well as mappings from O_1/O_2 to several intermediate ontologies IO_1, \dots, IO_k . Intermediate ontologies should have a significant overlap with the ontologies to be matched, i.e. the mappings should contain correspondences for a larger part of the ontologies' concepts. If possible it is reasonable to use the knowledge from different intermediate ontologies as they may complement each other. As composition of ontologies is likely very fast it is easily feasible to determine composed mappings for several intermediate ontologies.

In some cases (Fig. 2b), there is a centralized hub ontology HO that is predominant in the domain. Typically, such an ontology has many mappings to other ontologies. Any new ontology O_{new} can then be aligned with any other ontology O_1, \dots, O_n by first matching O_{new} to HO . Afterwards, the mapping $M_{O_{new},HO}$ can be composed with any available mapping M_{HO,O_i} in the domain. Hence, aligning the ontology O_{new} with any ontology O_i can be efficiently computed.

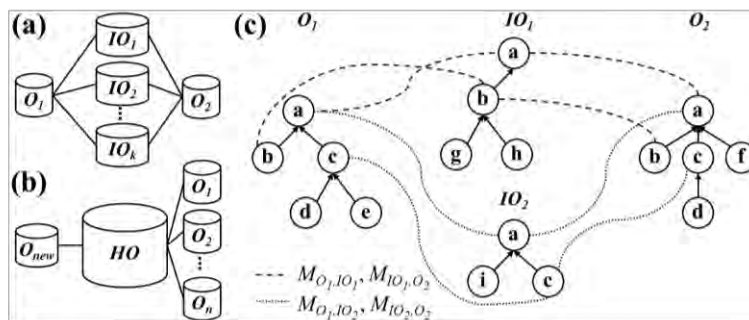


Figure 2. Mapping composition via intermediate ontologies IO_1, \dots, IO_n (a)
Match new ontology to hub ontology (b)
Example for composition-based ontology matching (c)

3.2 Operators

Previous research on the generic management of models and mappings [4] has already identified a series of operators that can be adapted for our purpose of ontology matching. In the following, we introduce the ontology and mapping operators match, compose and extract. Furthermore, we use merge to combine several (composed) mappings.

The match operator matches the concepts of an ontology O_A against the concepts of a second ontology O_B and directly determines an ontology mapping M_{AB} consisting of correspondences with similarity values (sim) between 0 and 1.

$$\begin{aligned} \text{match}(O_A, O_B): O_A \times O_B &\rightarrow M_{AB} \\ M_{AB} &= \{(c_1, c_2, sim) \mid c_1 \in O_A, c_2 \in O_B\} \end{aligned}$$

The compose operator allows for the composition of mappings, i.e., it combines two mappings (M_{AB}, M_{BC}) to indirectly determine a new mapping (M_{AC}). Two correspondences of different mappings can be composed to a new correspondence if the range concept of the first correspondence is equal to the domain concept of the second correspondence. Different functions can be used to aggregate the similarity values of correspondences ($aggSim$), e.g., by computing the average or maximum similarity.

$$\begin{aligned} \text{compose}(M_{AB}, M_{BC}): M_{AB} \times M_{BC} &\rightarrow M_{AC} \\ M_{AC} &= \{(c_1, c_2, aggSim(sim_1, sim_2)) \mid c_1 \in O_A, b \in O_B, \\ & c_2 \in O_C: \exists(c_1, b, sim_1) \in M_{AB} \wedge \exists(b, c_2, sim_2) \in M_{BC}\} \end{aligned}$$

The extract operator reduces an ontology O_A to a delta ontology ΔO_A by returning only those concepts that are not covered by an input mapping M_{AB} between O_A and another ontology O_B . It can be used to match only the delta ontologies ($\text{match}(\Delta O_A, \Delta O_B)$) to save the comparisons that are already covered by the (partial) mapping M_{AB} .

$$\begin{aligned} \text{extract}(O_A, M_{AB}): O_A \times M_{AB} &\rightarrow \Delta O_A \\ \Delta O_A &= \{c \mid c \in O_A, \nexists b \in O_B: (c, b, sim) \in M_{AB}\} \end{aligned}$$

The merge operator aggregates several input mappings between the same ontologies to a combined mapping. The merge decision is based on a minimum occurrence count occ in the k input mappings, i.e., a correspondence must appear in at least occ of the k input mappings. Note that $occ=1$ corresponds to a standard union whereas $occ=k$ corresponds to

the intersection of all mappings.

$$\begin{aligned} \text{merge}(M_{AB1}, \dots, M_{ABk}, occ): M_{AB1} \times \dots \times M_{ABk} \times occ &\rightarrow \\ & M_{AB} \\ M_{AB} &= \{(c_1, c_2, aggSim) \mid (c_1, c_2, sim) \text{ occurs in at least} \\ & occ \text{ mappings of } M_{AB1}, \dots, M_{ABk}\} \end{aligned}$$

3.3 Composition-Based Match Approach

The introduced operators are used within two algorithms that make up our composition-based match approach: *composeMatch* and *extendMatch*. *composeMatch* takes as input two ontologies O_1 and O_2 , a list of one or more intermediate ontologies IO_1, \dots, IO_k as well as the parameter occ denoting the occurrence count for mapping merge. The algorithm produces a composed mapping between O_1 and O_2 using the intermediate ontologies by reusing existing mappings. Firstly, for each intermediate ontology IO_i in the list we get the mappings between O_1 and IO_i as well as between IO_i and O_2 , e.g., from a repository. Afterwards the compose operator is applied to the mappings $M_{O_1, IO_i}, M_{IO_i, O_2}$ to determine a mapping between O_1 and O_2 . This composed mapping is added to the list of mappings (*MapList*). Finally, all mappings in *MapList* are merged to a combined mapping controlled by parameter occ . The merge of several mappings likely improves match quality. For example, the union of complementing intermediate ontologies can help to find more correct correspondences thereby improving recall. If the input list contains only one intermediate ontology, e.g., a known hub, the merge step can be omitted.

Typically, a composed mapping CM_{O_1, O_2} may not cover all parts of the ontologies O_1 and O_2 that need to be matched. Therefore, the algorithm *extendMatch* can be applied optionally to further improve recall and match quality. It takes the two ontologies as well as the composed mapping as input. To find additional correspondences between unmatched ontology parts we use the extract operator to determine the sub-ontologies of O_1 and O_2 that are not covered by CM_{O_1, O_2} . The resulting delta ontologies $\Delta O_1, \Delta O_2$ are matched directly using a specific match algorithm, e.g., string similarity of the attributes name and synonym. We then determine the union (merge with $occ=1$) of this direct mapping $DM_{\Delta O_1 \Delta O_2}$ and the composed mapping CM_{O_1, O_2} . Note that, all produced mappings can be filtered by selection

Algorithm *composeMatch*($O_1, O_2, IO_1 \dots IO_k, occ$)

Input: Two ontologies O_1 and O_2 , list of intermediate ontologies $IO_1 \dots IO_k$, occurrence count occ

Output: Composed mapping CM_{O_1, O_2}

```
1: MapList  $\leftarrow$  empty
2: for each  $IO_i \in IO$  do
3:    $M_{O_1, IO_i} \leftarrow$  getMapping( $O_1, IO_i$ )
4:    $M_{IO_i, O_2} \leftarrow$  getMapping( $IO_i, O_2$ )
5:   MapList.add(compose( $M_{O_1, IO_i}, M_{IO_i, O_2}$ ))
6: end for
7: return merge(MapList,  $occ$ )
```

criteria (e.g., a minimal similarity threshold) or advanced post-processing steps to improve precision, i.e., to reduce the number of incorrect correspondences.

Figure 2c illustrates an exemplary application of *composeMatch* for matching ontologies O_1 and O_2 via two intermediate ontologies IO_1 and IO_2 . Dotted lines represent the correspondences of O_1 and O_2 to the intermediate ontologies. The mapping composition (line 2-6) will produce the following *MapList* with two mappings between O_1 and O_2 consisting of two correspondences each: $\{(a,a), (b,b)\}, \{(c,c), (a,a)\}$. The merge aggregation of the *MapList* (line 7) with $occ=1$ results in the union mapping $\{(a,a), (b,b), (c,c)\}$ whereas $occ=2$ leaves only a single correspondence $\{(a,a)\}$. Not shown are the similarity values that need to be aggregated, e.g., by computing the average similarity.

4 Evaluation

4.1 Evaluation Setup

In all experiments, we align the Adult Mouse Anatomy ontology (MA) with the anatomical part of the NCI Thesaurus (NCIT). This match task is part of the annual OAEI contest so that the perfect mapping can be used for evaluating the quality (precision, recall and its combination F-measure) of the generated mappings. Mapping composition is performed with the help of four large intermediate ontologies, namely FMA [8], Uberon [25], RadLex [22], and UMLS [26] in their versions of late 2010. Table 1 summarizes statistical properties of the utilized ontologies and

Algorithm *extendMatch*(O_1, O_2, CM_{O_1, O_2})

Input: Two ontologies O_1 and O_2 ,
composed mapping CM_{O_1, O_2}

Output: Extended Mapping EM_{O_1, O_2}

```
1:  $\Delta O_1 \leftarrow$  extract( $O_1, CM_{O_1, O_2}$ )
2:  $\Delta O_2 \leftarrow$  extract( $O_2, \text{inverse}(CM_{O_1, O_2})$ )
3:  $DM_{\Delta O_1 \Delta O_2} \leftarrow$  match( $\Delta O_1, \Delta O_2$ )
4:  $EM_{O_1, O_2} \leftarrow$  merge( $\{CM_{O_1, O_2}, DM_{\Delta O_1 \Delta O_2}\}, 1$ )
5: return  $EM_{O_1, O_2}$ 
```

mappings. The ontologies significantly differ in their total number of concepts ($|C|$) and the number of name/synonym attributes per concept ($\emptyset\text{NameSyn}$) (Table 1a). All intermediate ontologies are significantly larger than MA and NCIT. The ontology mappings used for the algorithm *composeMatch* have been computed once based on the linguistic similarity (trigram with threshold 0.8) of concept names and synonyms. Hence we compose automatically determined mappings instead of manually verified ones making it more difficult to achieve high mapping quality. Table 1b reveals significant differences in the mapping coverage (Cov) and sizes ($|Map|$) for MA and NCIT. For UMLS and Uberon, the mappings cover up to 80% and more while RadLex is limited to about 40%, i.e. this intermediate ontology cannot provide correspondences for most concepts. The FMA mappings have only medium coverage potentially influenced by relatively few names and synonyms per concept (Table 1a) limiting the quality of linguistic matching. By contrast, Uberon is a promising intermediate ontology due to its high $\emptyset\text{NameSyn}$ value. We generally expect ontologies providing many synonym terms to be adequate intermediate ontologies w.r.t. linguistic matching.

Applying *extendMatch* determines the two Δ -ontologies: $\Delta O_1 = \{d, e\}$, $\Delta O_2 = \{d, f\}$ (line 1-2) which are then matched against each other (line 3). The resulting direct mapping, e.g., $DM_{\Delta O_1 \Delta O_2} = \{(d, d)\}$, is merged with the input mapping CM_{O_1, O_2} so that the final mapping $\{(a, a), (b, b), (c, c), (d, d)\}$ is obtained.

(a)

	C	∅ NameSyn
MA	2,738	1.1
NCIT	3,298	2.5
Uberon	4,958	4.9
UMLS	87,913	3.1
RadLex	30,773	1.6
FMA	81,059	1.8

(b)

Mapping	Cov _{Domain}	Cov _{Range}	Map
MA-Uberon	80%	45%	2300
Uberon-NCIT	33%	48%	1703
MA-UMLS	69%	3%	2975
UMLS-NCIT	5%	87%	4214
MA-RadLex	39%	3%	1082
RadLex-NCIT	4%	40%	1347
MA-FMA	57%	2%	1601
FMA-NCIT	3%	67%	2337

Table 1. Statistics for ontologies (a) and mappings (b)

The match operation within the *extendMatch* algorithm and the direct match computation consists of the steps pre-processing, similarity calculation, and post-processing. Pre-processing includes the elimination of delimiters and stop words, transformation to lower case letters, and word stemming. The similarity between ontology concept pairs is calculated based on the linguistic trigram similarity on concept names and synonyms. Post-processing consists of a MaxDelta selection [6] of correspondences returning for a concept the correspondences with the maximal similarity value or within a small delta distance to the maximal value. Furthermore, correspondences must meet a so-called CrissCross condition [12] for improved precision that eliminates conflicting correspondences (a_1, b_1) and (a_2, b_2) where a_2 is a child of a_1 but b_1 a child of b_2 or vice versa.

4.2 Composition-based matching

We first compare the quality of indirectly determined ontology mappings using the *composeMatch* algorithm as well as the impact of *extendMatch*. Fig. 3 summarizes the obtained mapping quality in terms of F-measure and compares them with the quality of a direct match (called as *no IO*). The direct matching based on linguistic similarity achieved a F-

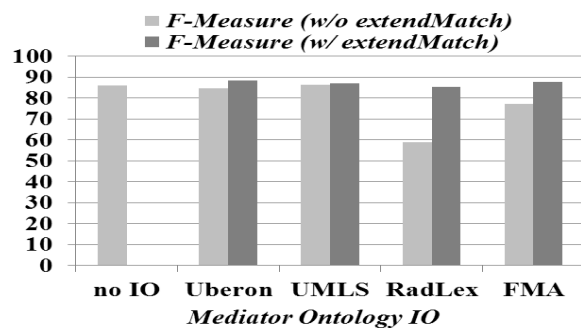


Figure 3. Compose via different intermediate ontologies

measure of 86% which is comparable to the best value of all previous OAEI contests (87.7%). The quality of the composed mappings (light grey bars) strongly depends on the utilized intermediate ontology and their associated mappings. The best F-measure values are achieved for composition via UMLS (86.2%) and Uberon (84.7%). Particularly, the UMLS-based mapping even exceeds the quality achieved by a direct match. Ontology mappings using FMA (77%) and RadLex (59%) only achieve a low quality influenced by the low coverage of their mappings to MA and NCIT. While RadLex is not primarily concerned with anatomy, Uberon provides a cross-species anatomy ontology and UMLS contains a huge anatomy part making these ontologies highly suitable for indirectly matching anatomy ontologies. The dark grey bars in Fig. 3 denote the achieved quality by an additional application of *extendMatch*. The results indicate that this additional step always leads to an improved quality. Interestingly, Uberon now achieves the best quality (88.2%) exceeding UMLS (87.0%) and the best OAEI result so far. This indicates that composition via Uberon finds non-trivial correspondences that cannot be identified by a direct match. The additional match effort of *extendMatch* improves match quality especially for intermediate ontologies with comparatively poor compose results (e.g., RadLex and FMA).

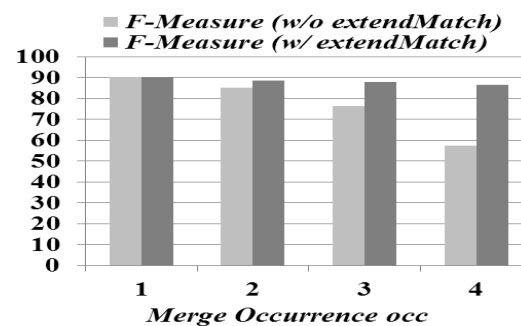


Figure 4. Merge composed mappings

Next, we determine whether the combination of several composed mappings from different intermediate ontologies improves match quality. Figure 4 shows the resulting F-measure values when merging the four composed mappings for different values for the occurrence count *occ* (specifying how often a correspondence has to occur in the individual composed mappings). The results show that merging several composed mappings improves match quality to up to 90.2% F-measure (recall 87.8%; precision 92.7%) for *occ*=1, i.e. when we take the union of the mappings. So we can outperform the quality of direct matching as well as the best OAEI result by our composition-based approach although we only compose automatically determined mappings. The intersection of the mappings (*occ*=4) turned out to be not effective (F-measure 57.4%) due to a significant reduction of recall, i.e. we can no longer take advantage of complementary correspondences provided by different intermediate ontologies. Additionally applying *extendMatch* leaves the result for *occ*=1 almost unchanged (90.3%) while it can significantly improve match qualities for larger *occ* values. Hence, the union of composed mappings can be used without applying an extra matching step. None of the previous approaches participating in OAEI anatomy track could exceed 87% F-Measure such that an increase to more than 90% is a significant improvement.

The execution times of the match process (without parsing ontologies/mappings) could be significantly reduced. The compose via all four intermediate ontologies and the following mapping merge took only 2.8s. The execution time for the additional *extendMatch* was 36s on average. By contrast the full direct match of the whole ontologies took 116s, i.e., the execution time could be reduced by up to a factor of 41 while achieving similar or even better match quality.

5 Related Work

The direct matching of large life science ontologies has been studied before [9, 15, 21]. Thereby different match approaches such as lexical and structural methods have been evaluated, e.g., in the domain of anatomy [28, 17].

The operators compose, match and extract

were introduced within the framework of model management [4]. They can be used in scenarios such as schema evolution to adapt dependent artifacts like instance data and views. In contrast we use these operators to efficiently match two ontologies based on composition.

The match compose operation has been introduced in schema matching before [3, 6] but was not applied for ontology matching. So far, there has been some attention on indirect matching and mapping composition in the life sciences. [27] derived indirect mappings using FMA as reference ontology. By contrast, we focus on using multiple complementing intermediate ontologies as well as an additional *extendMatch* to improve recall of compose. [24] presented an empirical analysis of mapping composition. They analyzed a pool of mappings without distinguishing different intermediate ontologies. Hence, it was not the focus to study which ontologies are useful hub ontologies. [2, 15] matched ontologies or other vocabularies by using a single ontology as domain/background knowledge. These strategies differ from our approach as they do not combine the knowledge of several different intermediate ontologies.

6 Conclusion and Future Work

We proposed a composition-based approach for indirectly matching life science ontologies via one or several intermediate ontologies. The goal is to reuse previously determined ontology mappings for improved match efficiency and quality. The approach is based on ontology and mapping operators compose, match, extract and merge. It allows the flexible combination of several composed mappings and the incremental extension of mappings by additional match steps for unmatched ontology concepts.

In our evaluation for large anatomy ontologies we considered four intermediate ontologies, namely UMLS, FMA, Uberon and RadLex. Overall, we achieved very good match quality (>90%) and significantly reduced execution times using a composition-based match instead of a direct match strategy. While the use of *extendMatch* is generally helpful to improve match quality, mapping composition alone was able to outperform the runtime and quality compared to direct matching especially when we merge several composed mappings.

Uberon and UMLS showed to be very effective intermediate ontologies and are thus suited as hub ontologies in the anatomy domain.

In future work, we plan to investigate composition-based ontology matching for further domains. We also want to study the impact of considering additional mappings, e.g. determined by structural matching or existing mappings from BioPortal. Furthermore, we want to investigate when it could be useful to compose more than two mappings within longer mapping chains.

Acknowledgments

This work is supported by the German Research Foundation (DFG), grant RA 497/18-1 (“Evolution of Ontologies and Mappings”).

References

1. Adult Mouse Anatomy: http://www.informatics.jax.org/searches/AMA_fm
2. Aleksovski, Z., Klein, M., Ten Kate, W., Van Harmelen, F.: Matching unstructured vocabularies using a background ontology, *Managing Knowledge in a World of Networks*, 182-197, Springer (2006)
3. Aumüller, D., Do, H.H., Massmann, S., Rahm, E.: Schema and ontology matching with COMA++. *Proc. SIGMOD* (2005)
4. Bernstein, P.A., Melnik, S.: Model management 2.0: manipulating richer mappings. *Proc. SIGMOD* (2007)
5. Bodenreider, O., Stevens, R.: Bio-ontologies: current trends and future directions. *Briefings in Bioinformatics* 7(3), 256-274 (2006)
6. Do, H.H. and Rahm, E.: COMA: a system for flexible combination of schema matching approaches. *Proc. VLDB* (2002)
7. Euzenat, J., Shvaiko, P.: *Ontology Matching*. Springer (2007)
8. Foundational Model of Anatomy: <http://sig.biostr.washington.edu/projects/fm/>
9. Ghazvinian, A., Noy, N.F., Musen, M.A.: Creating mappings for ontologies in biomedicine: Simple methods work. *Proc. AMIA Annual Symposium* (2009)
10. Gross, A., Hartung, M., Kirsten, T., Rahm, E.: On Matching Large Life Science Ontologies in Parallel. *Proc. Data Integration in the Life Sciences* (2010)
11. Jakoniene, V., Lambrix, P.: Ontology-based integration for bioinformatics. *Proc. ODBIS Workshop @ VLDB* (2005)
12. Jean-Mary, Y.R., Shironoshita, E.P., Kabuka, M.R.: Ontology matching with semantic verification. *Journal of Web Semantics* 7(3), 235-251 (2009)
13. Kirsten, T., Gross, A., Hartung, M., Rahm, E.: GOMMA: A Component-based Infrastructure for managing and analyzing Life Science Ontologies and their Evolution, Submitted to *Journal of Biomedical Semantics* (2011)
14. Lambrix, P., Edberg, A.: Evaluation of ontology merging tools in bioinformatics. In: *Proc. of the 8th Pacific Symposium on Biocomputing* (2003)
15. Lambrix, P., Tan, H.: SAMBO - A system for aligning and merging biomedical ontologies. *Journal of Web Semantics* 4(3), 196-206 (2006)
16. Lambrix, P., Tan, H., Jakoniene, V., Strömbäck, L.: *Biological Ontologies. Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences* (2007)
17. Mork, P., Bernstein, P.A.: Adapting a Generic Match Algorithm to Align Ontologies of Human Anatomy. *Proc. ICDE* (2004)
18. Mouse Models of Human Cancers Consortium: <http://www.nih.gov/science/models/mouse/resources/hcc.html>
19. NCI Thesaurus: <http://ncit.nci.nih.gov/>
20. Noy, N.F. et al.: BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research* 37 (Web Server Issue), W170-W173 (2009)
21. Ontology Alignment Evaluation Initiative: <http://oaei.ontologymatching.org>
22. RadLex: <http://www.radlex.org>
23. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. *VLDB Journal* 10(4), 334-350 (2001)
24. Tordai, A. et al.: Lost in Translation? Empirical Analysis of Mapping Compositions for Large Ontologies. *Proc. Ontology Matching Workshop @ ISWC* (2010)
25. Uber Anatomy Ontology: http://obofoundry.org/wiki/index.php/UBERON:Main_Page
26. Unified Medical Language System: <http://www.nlm.nih.gov/research/umls>
27. Zhang, S., Bodenreider, O.: Alignment of multiple ontologies of anatomy: Deriving indirect mappings from direct mappings to a reference. *AMIA Annual Symposium* (2005)
28. Zhang, S., Bodenreider, O.: Experience in aligning anatomical ontologies. *Semantic Web and information systems* 3(2), 1-26 (2007)