

Interactive vocabulary alignment

Jacco van Ossenbruggen^{1,2}, Michiel Hildebrand¹, and Victor de Boer¹

¹ VU University Amsterdam

² CWI Amsterdam, The Netherlands

Abstract. In many heritage institutes, objects are routinely described using terms from predefined vocabularies. When object collections need to be merged or linked, the question arises how those vocabularies relate. In practice it often unclear for data providers how well alignment tools will perform on their specific vocabularies. This creates a bottleneck to align vocabularies, as data providers want to have tight control over the quality of their data. We will discuss the key limitations of current tools in more detail and propose an alternative approach. We will show how this approach has been used in two alignment use cases, and demonstrate how it is currently supported by our Amalgame alignment platform.

1 Introduction

In the library, archive, museum and many other domains, objects are routinely described using terms from predefined vocabularies. When object collections need to be merged or linked, a typical question that needs to be answered is how those vocabularies relate. More specifically, one would like to know which concepts from different vocabularies correspond to one another. We will call a set of such correspondences an *alignment*.

There is an active research field that studies methods and techniques to generate alignments automatically. We experienced that in practice it is, however, difficult to apply these techniques to vocabularies in the cultural heritage domain. Most alignment tools are not designed for the large but shallow vocabularies typical in this domain. Furthermore, tools provide little support to analyse large sets of correspondences, making it difficult to assess the quality of the generated results. To tackle these issues we propose an interactive approach to vocabulary alignment.

In the next section we discuss the limitations of fully automatic alignment tools. In Sect. 3 we describe a semi-automatic, interactive approach. In Sect. 4 we will show how this approach has been used in two alignment use cases, and demonstrate how it is currently supported by our *Amalgame* alignment platform³. Finally, we reflect on our approach and discuss future work.

³ Amalgame is open source and available at <http://semanticweb.cs.vu.nl/amalgame/>. All alignment strategies discussed in the use cases have been published at <http://semanticweb.cs.vu.nl/lod/tpdl2011/> and can be “replayed” in Amalgame, allowing full replication of all alignments described in this paper.

2 Problem analysis

There is an active research field that studies methods and techniques to generate alignments automatically, and the tools produced by this field are evaluated yearly in the context of the Ontology Alignment Evaluation Initiative (OAEI)⁴. A key insight from this field is that two concepts can be similar or dissimilar along many different dimensions [3]. Automatically finding similar concepts typically requires some hybrid approach that combines different techniques, each addressing a part of the total set of potentially interesting dimensions. Another important insight is that the application context in which the alignment will be deployed often influences what constitutes a “good” alignment [5]: two concepts might be regarded as sufficiently similar in one context, but not in another. The main approach in vocabulary alignment is to develop hybrid tools that try to fully automatically find some smart combination of techniques to generate an alignment, and (b) allow the developer to tune the tool so that the alignment fits a specific application context.

While the approach sketched above is well established, both results from our previous work [9, 7, 8] and feedback received from domain experts during our work in the MultimediaN E-Culture⁵, Europeana(Connect)⁶ and PrestoPrime projects⁷, indicate that it also has some major limitations when it has to be applied in the cultural heritage field.

First, domain experts find it hard to determine how well a tool would perform for their alignment task. From the alignment research literature, it is clear how each tool performs on the data used in the evaluation experiments. However, due to the complexity of the good performing tools, it often remains unclear *why* some tools perform better than others, so it is hard for experts to predict which tool would be suitable for their own data set.

Second, experts perceive the current tools to not support the large and shallow vocabularies that are typical for their domain. Most alignment tools target complex vocabularies with different ontological relations, but only several 100s or 1000s of classes. In the cultural heritage domain the vocabularies typically contain only a few thesaurus relations, but frequently contain over 10,000s or even 100,000s of concepts. When run on larger vocabularies, many tools simply crash, or fail to finish alignment runs within a reasonable amount of time.

Third, when a tool finishes successfully, it typically produces a result set with a large number (e.g. over 100k) of correspondences, but provide little support to assess the quality of these results. Furthermore, the quality of the correspondences might not be homogeneously distributed across the alignment result set. Different subsets of alignments might have different features that determine the quality of the end result. Transparent and interactive assessment is crucial to be able to decide whether the result is of sufficient quality.

⁴ <http://oaei.ontologymatching.org/>

⁵ <http://e-culture.multimediam.nl/>

⁶ <http://www.europeanaconnect.eu/>

⁷ <http://www.prestoprime.org/>

Fourth, when the results are not sufficient it is unclear how the tool should be (re-)configured to improve the results. Experts need to be able to understand why a tool found erroneous correspondences and how to get rid of them in a next step to improve precision. When the tool failed to find correct correspondences, the experts need to know how to find those in a next step to improve recall. This often requires insight in how the alignment algorithms work, and how to configure them to adjust them to the specific needs of vocabularies at hand.

Remark that the first two problems are related to the fact that fully automatic alignment tends to result in complex techniques that are relatively slow on large data sets and hard to explain to domain users. The last two problems are due to the fact that current tools are designed to produce an alignment of sufficiently high quality in a single run, without much input from the user, while in practice experts feel that the required quality can only be achieved by multiple runs, where each run requires their input.

In the next section, we sketch an alignment approach that is based on these insights. We then show the feasibility of our approach by discussing two use cases of vocabulary alignments in which we have used this approach.

3 The Amalgame approach to vocabulary alignment

To address the problems above, we developed an alignment approach that improves the speed and transparency of the alignment process by drastically reducing the complexity of the technology, allowing the user to combine a limited number of basic building blocks into an alignment workflow targeted to the data set at hand. Each building block should be sufficiently simple to produce an understandable result. Which blocks to use and in what order or combination is fully controlled by the user. Furthermore, produced alignments (both intermediate and end results) can be easily evaluated to give insight in their quality.

We have built a prototype alignment service that has been designed with this approach in mind, and used the prototype to create alignments in two different use cases, that will be discussed in the next section. Here we sketch an high level overview of the Amalgame alignment methodology and will flesh out some interesting details in the context of the use case descriptions.

3.1 Vocabulary analysis

An assumption of the interactive approach is that the user has knowledge of the vocabularies being aligned. Here, we focus on vocabularies that can be represented by SKOS [6]. For such SKOS-like vocabularies we identify two types of characteristics. First, the user needs to know how the vocabularies differ in size and heterogeneity. Second, the user has to identify the concepts' properties that can be used in string matching. Third, the user has to identify other properties that can be matched, such as hierarchical and associative relations.

3.2 Workflow components

Our approach is to have the user interactively construct an alignment workflow. The individual building blocks of this workflow consist of: **selectors** to define which concepts to use from the source and target vocabularies, **matchers** to find correspondences between the selected source and target concepts, **partitioners** to split sets of correspondences, **mergers** to create unions of specific subsets, **analyse tools** to investigate the mappings, and **filters** to select specific correspondences and discard others.

3.3 Interactive alignment

Alignment within Amalgame is a process where the user iteratively applies matchers, partitions the result set, and applies new matchers or a filter. After each step the user typically analyzes the results to determine the next step. We identify five typical scenarios, depending on the outcome of the analysis.

- The first scenario is that a user decides the results are no good at all, in which case all results are simply discarded after analysis. Assuming the technique used is sufficiently simple, the user will understand from the analysis what caused the failure and will be able to try another matching run, using another technique or a better configuration of the technique used in the previous run.
- The second scenario is that the results are good, but that recall is low. To improve recall, the user can proceed by matching only the concepts that have not yet been aligned. Note that this result set is typically a smaller set, so the user may decide to deploy computationally more expensive matching techniques to improve recall in subsequent runs.
- The third scenario is that the results are good, but that precision is low. To improve precision, users need to find filters that allow them to distinguish true from false correspondences. Again, more expensive techniques can be used to boost precision for smaller subsets.
- The fourth scenario is that a user decides that the results are of sufficient quality, after which she exports them to the desired format and we consider the alignment task to be successfully finished.
- The fifth scenario is that the user finds the results of insufficient quality, but is out of options and does not know how they can be further improved, in which case we consider the alignment task to be failed.

In practice, we found the first scenario useful to quickly try some alternative matchers, and to compare, analyse and discard the results, just to develop some intuition before the real alignment task starts. Many alignment tasks, including the first two use cases discussed below, are based on an iteration of the second and third scenario. Ideally, with each iteration the set of concepts that have to still be aligned (to improve recall) and the set of correspondences that still have to be filtered (to improve precision) decreases, or, if not, the user gains some knowledge to achieve this in the next step.

4 Use cases

In this section we describe two alignment use cases. We found that, in practice, the in-house vocabularies from different institutes are sometimes directly aligned with each other, but typically they are indirectly related by aligning them to the same external vocabulary. As the first use case we explore such an alignment of an in-house vocabulary to an external vocabulary. We consider the alignment of the thesaurus of the Netherlands Institute for Sound and Vision, GTAA, with a general linguistic vocabulary of Dutch, Cornetto. A benefit of an alignment with such an external vocabulary is that this also makes the alignments of this vocabulary available for the in-house vocabulary. For example, Cornetto already contains links to the English WordNet. A different example where alignment is required, is when a new version of a vocabulary is released, and no direct links between the two are maintained. In the second use case we consider the mapping of two different versions of WordNet. The two use cases show typical examples of one-to-one, one-to-many and many-to-many correspondences (abbreviated as $1-1$, $1-n$ and $n-m$ below). While our approach could be applied to a wide variety of mapping relations, the use cases focus on relatively simple, bi-directional equivalence relations. More complex relationships, e.g. as described in [2], could be addressed by either deploying more complex workflows or more manual intellectual input.

4.1 In-house to general: GTAA to Cornetto

The Netherlands Institute for Sound and Vision uses an in-house thesaurus for the documentation of audiovisual content. This so-called GTAA thesaurus (Dutch acronym for Common Thesaurus Audiovisual Archives) contains approximately 160,000 terms in six facets: subjects, locations, person names, organization names, maker names and genres. In this use case we focus on the terms in the subjects facet.

Cornetto is a WordNet-like lexical semantic database of Dutch that contains 70,000 synsets [10]. Compared to the GTAA subject terms, the synsets provide a large number of additional synonyms and an extended description. The synsets are linked into an elaborate hierarchical structure.

To goal of making the alignment is to improve Dutch access to the institute's collection by taking advantage of Cornetto's additional labels (e.g. synonyms) and semantic relations to GTAA's subject terms. In addition, the existing alignment between Cornetto and WordNet could also provide an English access point to the archive.

For this use case we map the GTAA subject terms to Cornetto synsets. As Cornetto contains the same words in different synsets (e.g. homonyms), we can expect that string matching techniques will find multiple synsets for many GTAA subject terms. Our focus is to choose the right target synset(s) for each source. Typically, this will be one synset per GTAA subject term (that is, $n-1$ correspondences), but there might be cases where multiple synsets are good

candidates. In this case, the aim is to find not the best, but all correct targets (that is, n - m correspondences).

Vocabulary analysis We start the alignment process with an exploration of the GTAA subject terms. In total there are 3,932 subject terms. All terms have at least one preferred label, often an alternative label and one or more related terms, and some have a description. In addition, the subjects are organized in an hierarchical structure. We observe that the majority of the terms are nouns. In Cornetto this part of speech distinction is explicit, as each synset is of word type: noun (52,845), verb (9,017), adjective or adverb. Ideally, we would like to map the nouns in GTAA to the nouns in Cornetto. However, there is no explicit information in GTAA to automatically distinguish the nouns from the verbs. We choose the next best solution and start with the alignment of all GTAA subject terms to the nouns in Cornetto. We assume that there will be no or very few verbs from GTAA that will be incorrectly mapped to the nouns in Cornetto.

We also observe that the most labels of the GTAA subject terms are in plural form, whereas the labels in Cornetto are in singular form. When matching the labels we should account for this difference. Finally, we observe that where GTAA discriminates between preferred and alternative labels, Cornetto only has one type of label, which has been mapped to `skos:altLabel`.

Interactive alignment Given the discussions above, it is not *a priori* clear which string matching strategy to use. We expect that using alternative labels, in addition to the preferred labels, will increase recall, but are unsure at what expense (in terms of precision). Similarly, we expect that stemming will deal with the plural GTAA nouns and singular Cornetto nouns, but it might also introduce new problems. We decide to explore different options and try matching including and excluding GTAA alternative labels. We also match with and without stemming.

	Preferred labels			Preferred + alternative labels		
	total	n-1	n-m	total	n-1	n-m
exact	1,190 (30%)	880	310	1,319 (33%)	829	490
stem	2,493 (63%)	1785	708	2,725 (69%)	1655	1070

Table 1. Number of correspondences between GTAA and Cornetto. Horizontally, the labels used: preferred labels only and including alternative labels. Vertically, the label similarity metric: exact matching or matching after stemming.

Table 1 shows the statistics for the different string matching techniques⁸. From the column labeled *total*, we observe that there is indeed a large increase when stemming is used. We can also observe that by including the alternative labels more correspondences are found. Based on these observation we might opt

⁸ The mappings generated in this use case can be found online at http://semanticweb.cs.vu.nl/lod/tpd12011/gtaa_cornetto

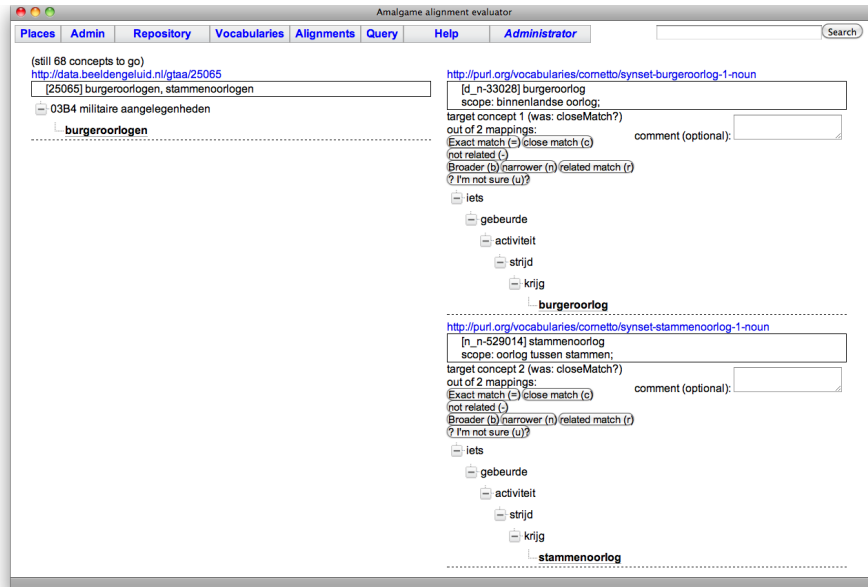


Fig. 1. Screenshot of the Amalgame evaluation prototype. On the left the source concept from GTAA and on the right two target concepts from Cornetto. The GTAA concept “burgeroorlog” (dutch for civil war) with alternative label “stammenoorlog” (dutch from tribal war) is mapped to two different targets. The target concepts in Cornetto, civil war and tribal war, are siblings as they are two specific types of war.

for the approach that gives us the highest recall: matching the stems of both the preferred and alternative labels. Before we make this decision there is, however, another important characteristic of the results that we should consider. How many target concepts are found for each source concept? And in case multiple targets are found, is this caused by ambiguity of the source concept or are all targets valid alternatives?

To investigate different types of alignments we use Amalgame to partition the set of correspondences. We partition them in a set where the source concepts have only 1 target, and another set where the source concepts have multiple targets. Table 1 lists the number of sources that are mapped to only 1 target (the $n-1$ column includes both $1-1$ and true $n-1$ results, and mapped to multiple targets (idem, $n-m$ also includes $1-m$). We observe that the number of $n-1$ alignments is larger when only the preferred labels are included. In other words, the alternative labels primarily introduce extra targets for sources that were already mapped. Do these alignments introduce unnecessary ambiguity, or are the additional targets valid alternatives?

To analyze the results in more in detail we use Amalgame to visualize correspondences including the relevant information of the source and target concepts.

In this case, we are interested in the n - m mappings introduced by the alternative labels. We produce this set by subtracting the 708 n - m correspondences found by matching preferred labels from the larger set of 1070 n - m found by matching both preferred and alternative labels. From the resulting set we take a random sample of 25 correspondences to investigate in detail. Figure 1 shows a screen shot of this investigation. For a single source concept it lists the multiple target concepts. In addition, all alternative labels, descriptions and related terms are shown. Going through the sample set we found four different types of n - m correspondences:

1. One of the targets is more generic than the others. Cornetto is more fine-grained than GTAA. A single concept in GTAA containing multiple labels, e.g. “poison, pesticide”, is mapped to different targets in Cornetto, where “poison” is more generic than “pesticide”. In this case we want to select the most generic term. Optionally, we could create narrower matches between the other targets, but this is outside the scope of this paper.
2. The targets are siblings of each other. Again the granularity difference between the vocabularies often causes a single concept in GTAA, e.g. “civil war and tribal war”, to be mapped to different targets in Cornetto, where “civil war” and “tribal war” are siblings as they are more specific types of “war” (shown in Figure 1). In this case all targets are valid alternatives and we want to keep an 1- n correspondence to all siblings.
3. The targets are about the same topic. Some concepts in GTAA contain labels for different types of things, but related in topic e.g. “beekeeping and honey combs”. In Cornetto these are different terms in completely different parts of the hierarchy. We choose again to keep all targets and create a 1- n correspondence. If we would have the rights to modify GTAA, we could also decide to split the source concepts into two separate concepts.
4. The targets are different senses of the source concept. A GTAA concept is matched to one concept from Cornetto by its preferred label and to another by its alternative label. For example, by the preferred label “capitulate” a single concept from Cornetto is found. By the alternative label “surrender” it finds the same concept, but also the concept that refers to “surrender of attention”. In this case the source concept is ambiguous and only one target should be selected.

We conclude that by using only the preferred labels valid alternatives are excluded. Therefore, we choose to include alternative labels and match them after stemming. The n -1 correspondences generated with this configuration are likely to be correct, as we used a simple matching algorithm that fits well with the labels in our vocabularies. Evaluation of a random sample of 25 confirms this assumption, as all correspondences are indeed correct. At the other hand we have a larger set of n - m correspondences. The analysis of this set provided us with a number of different cases. How can we use this knowledge to find the valid n - m correspondences and, in case of ambiguity, select the best candidate to get the n -1 correspondence we are looking for?

To automatically detect different types of correspondences and select the best target candidates Amalgame provides a number of strategies. We configure these strategies for the different types of n - m correspondences. We start with the n - m set (1070 source concepts) and try to identify the correspondences for each case. For 91 source concepts we can find a target that is more generic than the other targets. These concepts are found by configuring the Amalgame partitioning component to check for hierarchical relations between the targets. From the remaining correspondences, 72 sources have sibling targets.

For the remaining n - m correspondences, we try to automatically detect the most suited candidate. We observed that the wrong targets can occur in different sub-trees of Cornetto. Therefore, we can identify the best target by the hierarchical similarity to the source target. For each ambiguous correspondence we check if the source and target have similar ancestors or descendants. To test for similarity between the terms in the hierarchy we use as a base set the n -1 correspondences. When the hierarchy of one target has more aligned concepts with the hierarchy of the source it is a better candidate. As this method adds new correspondences, it extends the base set, possibly relevant for further disambiguation. Therefore, we repeat this procedure until no more additional matches are found. In total, for 342 source concepts we manage to find a distinguishing target.

Finally, we decide to align all remaining GTAA subjects to the verbs in Cornetto. Analysis of the vocabularies also makes clear that the labels of the verbs, in both vocabularies, are in infinitive form. Therefore, we choose to align them using exact string matching. For 115 source concepts we find correspondences, 78 of these are n -1 mappings, while 37 are n - m mappings. As the n - m set is very small, we can manually evaluate it. Within 14 minutes we manually disambiguated 19 sources, and accepted multiple alternatives for two sources. For the remaining 14 source concepts we decided they were falsely mapped. All were nouns that were not mapped due to limitations of the stemming algorithm. We expect the same stemming problem causes errors in the set of n -1 correspondences, and also manually evaluate these. Within only 5 minutes we found the 13 source concepts were it went wrong.

Results In total we found matches for 2275 (58%) concepts from the GTAA subjects facet. From these the large majority (2160, 55%) were matched to Cornetto nouns. For 42% of the GTAA subjects we found a correspondence to only one target. As we used a simple matching technique, we expected high precision for this subset. In an evaluation of a small sample of this set all correspondences were judged to be correct. In the remaining set we identified four ways in which multiple targets were found. We configured the filter components to identify these cases. For more than half of the 1-n matches we managed to either select the best target or confirm that all targets are valid alternatives. To judge the other half of the matches manual evaluation is required. In future work we would like to perform such an evaluation with the users of GTAA. Finally, only 115 GTAA subject terms were mapped to Cornetto verbs. This small set we manually evaluated in only a few minutes.

4.2 Versioning: WordNet 3.0 to WordNet 2.0

WordNet is a large lexical database of English published by Princeton University. It groups nouns, verbs, adjectives and adverbs into sets of cognitive synonyms (synsets), each expressing a distinct concept [4]. W3C released an RDF version of Princeton’s WordNet 2.0 in June 2006 [1]. In August 2010 we released an RDF conversion of Princeton’s WordNet 3.0 as Linked Open Data. Until now, there is no reliable data set that specifies which synset in version 3.0 correspond to which synset in version 2.0. A typical goal of creating these correspondences would be to update corpora indexed with the old version to the new version.

Vocabulary analysis To be able to treat WordNet as a SKOS vocabulary, we use a simple schema mapping: WordNet synsets are mapped to SKOS concepts, WordNet sense labels to SKOS altLabels and WordNet glosses to SKOS definitions. WordNet 2.0 consists of 115,424 concepts with a total of 203147 labels. WordNet 3.0 has slightly more (117,657) concepts with 206,976 labels.

Because WordNet maintenance is largely a manual effort, we expect many concepts will have remained the same and will be easy to map. Concepts that we will choose to leave unmapped are those 2.0 synsets that have been dropped in the new version, without having a counterpart in the new version and the 3.0 synsets that are newly added without having a counterpart in the old version. Concepts that we would like to map but could be hard to do automatically include concepts that have splitted or merged between versions, and concepts of which so many properties have changed that it is hard to tell if we are dealing with the “same” concepts or not.

Both vocabularies are splitted into nouns (70%), verbs (12%), adjectives (15%) and adverbs (3%). We assume that by mapping only nouns to nouns, verbs to verbs, etc. we can both reduce the search space and avoid many erroneous mappings between homonyms in different parts of speech. This approach risks missing concepts that moved to another part of speech category, but we assume this to occur very infrequently or not at all.

Interactive alignment When aligning WordNet 3.0 to 2.0 we would like to explicitly use our knowledge of the fact that we are aligning two versions of the same vocabulary. For example, given the large amount of homonymy, we expect a simple label match to produce many correspondences, most of which will be wrong. In contrast, we expect the definitions to be unique for most concepts, and since manually updating many definitions is hard manual work, we expect the majority of the concepts to have the same definition in both versions.

So as a first step, we try a quick case insensitive match on skos:definition. Selecting only the *1-1* mappings results leaves us with 103,521 correspondences (set 1a⁹), covering already 89.7% of all 2.0 synsets. Of the *n-m* correspondences, 931 can be reduced to *1-1* (set 1b) by simply matching also the labels. We quickly evaluate the remaining 26 correspondences (set 1c) manually, and conclude these are all cases with duplicate synsets in one or both versions, so all the 26 remaining *n-m* correspondences turn out to be correct too. After this simple first step, we

⁹ See online results at <http://semanticweb.cs.vu.nl/lod/tpd12011/wn3020>

only need to align less than 10% of the original number of concepts, so we can afford more expensive techniques in the following steps.

As a second step, we run a cheap, case insensitive label match on the remaining concepts. This yields another 6379 *1-1* correspondences (set 2a), which we assume to be mostly correct. As expected, it also results in a relatively large number of *n-m* correspondences: 8528 matches between only 3502 source and 3319 targets. In this set we thus expect many wrong homonym matches. We run a more expensive string distance matcher on the definitions, after which we select, for each 2.0 target concept, the source with the most similar definition. This reduces the set to 3310 mappings between 2807 sources and 3310 targets, for only 9 targets we find 19 mappings to two or more equally similar sources. A quick manual evaluation found that only 8 of these 19 were correct (set 2b). Repeating this step in the other direction, by selecting for each of the 2807 source the most similar target, we find 2800 *1-1* mappings (set 2c), with 14 mappings for the 7 sources for which there two equally similar targets. Manual evaluation found 9 of these correct (set 2d).

Results We have created three distinct subsets of correspondences in the first step and four subsets in the second step. Together, these seven sets consist of 113,675 correspondences for a similar number of WordNet 2.0 concepts, covering 98.48% of all 2.0 synsets. For each subset, we can easily describe how it has been created, and why we would or would not trust the correspondences they contain. A more thorough manual evaluation could take this into account, by taking strategic samples from each subset. The coverage can be further increased by trying to map concepts for which (all) the labels have been changed between versions, as happens when spelling errors are detected or new spelling conventions are applied, but this is out of scope for this paper.

5 Discussion

We conclude it is feasible to construct an alignment workflow for relatively large SKOS-like vocabularies by combining simple techniques. With the prior knowledge of the vocabularies and analysis of the correspondences we iteratively increased recall and precision. The resulting alignments are comprised of multiple homogeneous subsets of correspondences. This allows for targeted evaluation per subset. In addition, this allows to combine evidence from multiple subsets to increase precision, or strategically select multiple subsets to increase recall.

A potential drawback of our approach is that the selection, configuration and combination of components is the responsibility of the user. This makes the approach less attractive for data sets where fully automatic approaches produce results of sufficient quality. A potential risk is that we assume a finite and relatively small set of basic components. Amalgame currently provides a number of such components, some of these were used across use cases. During the specific use cases, however, we also found a need for additional components. Creating these components was straightforward. New use cases might require new components as well.

The workflows for the use case presented in this paper were created by the authors, using an experimental interface. Our longer term goal is to support vocabulary owners to create their own alignments. This requires a user interface to iteratively construct alignment workflows. Currently we are developing such a user interface. The interface combines the construction of a workflow, with the analysis of mappings. Thus, each time extending a single node and using the analysis tools to investigate intermediate results. In future work we will evaluate such an interface with the vocabulary owners.

Acknowledgements

We thank W. van Hage, A. Isaac, C. Reverté Reverté, A. Tordai and J. Wielemaker for their feedback and help in the development of Amalgame. M. van Assem produced the RDF conversions for WordNet 2.0 and 3.0. This work was partially supported by the PrestoPRIME and EuropeanaConnect project.

References

1. van Assem, M., Gangemi, A., Schreiber, G.: Conversion of WordNet to a standard RDF/OWL representation. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06). Genoa, Italy (May 2006), <http://www.cs.vu.nl/~mark/papers/Assem06a.pdf>
2. Doerr, M.: Semantic problems of thesaurus mapping. *J. Digit. Inf.* 1(8) (2001)
3. Euzenat, J., Shvaiko, P.: *Ontology matching*. Springer-Verlag, Heidelberg (DE) (2007)
4. Fellbaum, C. (ed.): *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication Series, MIT Press (1998)
5. van Hage, W.R., Isaac, A., Aleksovski, Z.: Sample evaluation of ontology-matching systems. In: Garcia-Castro, R., Vrandečić, D., Gómez-Pérez, A., Sure, Y., Huang, Z. (eds.) *EON. CEUR Workshop Proceedings*, vol. 329, pp. 41–50. CEUR-WS.org (2007)
6. Miles, A., Bechhofer, S.: Skos simple knowledge organization system reference. W3C Recommendation (August 18 2009), <http://www.w3.org/TR/skos-reference/>
7. Tordai, A., van Ossenbruggen, J.R., Ghazvinian, A., Musen, M.A., Noy, N.F.: Lost In Translation? Empirical Analysis Of Mapping Compositions For Large Ontologies. In: *Proceedings of International Workshop on Ontology Matching 2010* (5). CEUR-WS (November 2010)
8. Tordai, A., van Ossenbruggen, J.R., Schreiber, G., Wielinga, B.: Aligning Large SKOS-Like Vocabularies. In: *Proceedings of European Semantic Web Conference 2010* (7). pp. 198 – 212. *Lecture Notes in Computer Science*, Springer (May 2010), http://dx.doi.org/10.1007/978-3-642-13486-9_14
9. Tordai, A., van Ossenbruggen, J., Schreiber, G.: Combining vocabulary alignment techniques. In: *K-CAP '09: Proceedings of the fifth international conference on Knowledge capture*. pp. 25–32. ACM, New York, NY, USA (2009)
10. Vossen, P., Maks, I., Segers, R., van der Vliet, H.: Integrating lexical units, synsets and ontology in the Cornetto database. In: (ELRA), E.L.R.A. (ed.) *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)* (2008)