WeSeE-Match Results for OEAI 2012

Heiko Paulheim

Technische Universität Darmstadt
paulheim@ke.tu-darmstadt.de

Abstract. WeSeE-Match is a simple, element-based ontology matching tool. Its basic technique is invoking a web search engine request for each concept and determining element similarity based on the similarity of the search results obtained. Multi-lingual ontologies are translated using a standard web based translation service. The results show that the approach, despite its simplicity, is competitive with the state of the art.

1 Presentation of the system

1.1 State, Purpose, and General Statement

The idea of *WeSeE-Match* is to use information on the web for matching ontologies. When developing the algorithm, we were guided by the way a human would possibly solve a matching task. Consider the following example from the OAEI anatomy track[1]: one element in the reference alignment are the two classes with labels *eyelid tarsus* and *tarsal plate*, respectively. As a person not trained in anatomy, one might assume that they have something in common, but one could not tell without doubt.

For a human, the most straight forward strategy in the internet age would be to search for both terms with a search engine, look at the results, and try to figure out whether the websites returned by both searches talk about the same thing. Implicitly, what a human does is identifying relevant sources of information on the web, and analyzing their contents for similarity with respect to the search term given. This naive algorithm is implemented in *WeSeE-Match*.

1.2 Specific Techniques Used

The core idea of our approach is to use a web search engine for retrieving web documents that a relevant for concepts in the ontologies to match. For getting search terms from ontology concepts (i.e., classes and properties), we use the labels, comments, and URI fragments of those concepts as search terms. The search results of all concepts are then compared to each other. The more similar the search results are, the higher the concepts' similarity score.

To search for websites, we use the Microsoft Bing Search API[2]. We use URI fragments, labels, and comments of each concept as search strings, and perform some preprocessing, i.e., splitting camel case and underscore separated words into single words,

[1] http://oaei.ontologymatching.org/2012/anatomy/
[2] http://www.bing.com/toolbox/bingdeveloper/

and omitting stop words. While the approach itself is independent of the actual search engine used (although the results might differ), we have chosen Bing to evaluate our approach because of the larger amount of queries that can be posed in the free version (compared to, e.g., Google).

For every search result, all the titles and summaries of web pages provided by the search engine are put together into one *describing document*. This approach allows us to parse only the search engine's answer, while avoiding the computational burden of retrieving and parsing all websites in the result sets. The answer provided by the Bing search engine contains titles and excerpts from the website (i.e., some sentences surrounding the occurance of the search term in the website). Therefore, we do not use *whole* websites, but ideally only *relevant parts* of those web sites, i.e., we exploit the search engine both for information retrieval and for information extraction.

For each concept c, we perform a single search each for the fragment, the label, and the comment (if present), thus, we generate up to three documents $doc_{fragment}(c)$, $doc_{label}(c)$, and $doc_{comment}(c)$. The similarity score for each pair of concepts is then computed as the maximum similarity over all of the documents generated for those concepts:

$$sim(c_1, c_2) := max_{i,j \in \{fragment,label,comment\}} sim^*(doc_i(c_1), doc_j(c_2)) \quad (1)$$

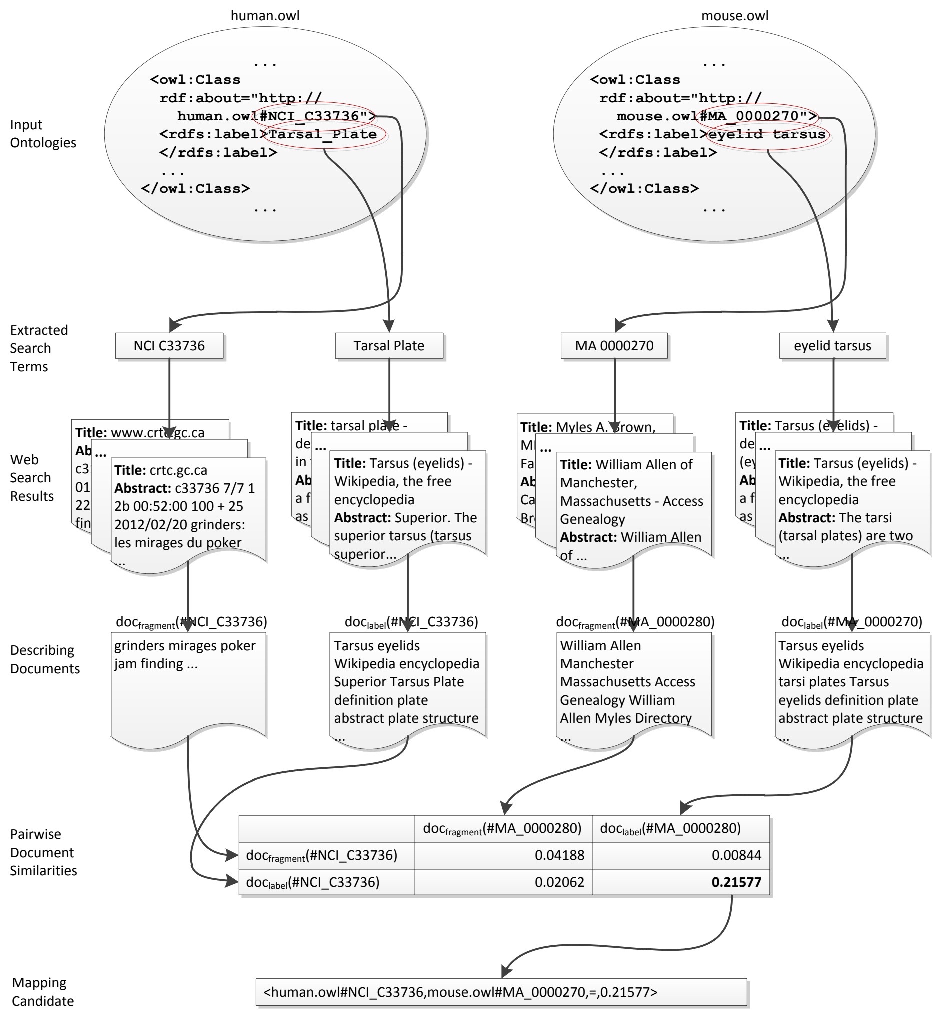
For computing the similarity sim^* of two documents, we compute a TF-IDF score, based on the complete set of documents retrieved for all concepts in both ontologies.

Using the TF-IDF measure for computing the similarity of the documents has several advantages. First, stop words like *and*, *or*, and so on are inherently filtered, because they occur in the majority of documents. Second, terms that are common in the domain and thus have little value for disambiguating mappings are also weighted lower. For example, the word *anatomy* will occur quite frequently in the anatomy track, thus, it has only little value for determining mappings there. On the other hand, in the library track, it will be a useful topic identifier and thus be helpful to identify mappings. The TF-IDF measure guarantees that the word *anatomy* gets weighted accordingly in each track.

The result is a score matrix with elements between 0 and 1 for each pair of concepts from both ontologies. For each row and each column where there is a score exceeding τ, we return that pair of concepts with the highest score as a mapping. Since most ontology matching problems only look for $1:1$ mappings, we optionally use edit distance for tie breaking if there is more than one candidate sharing the maximum score. This happens, for example, for pairs like *Proceedings – Proceedings* and *Proceedings – InProceedings* in the conference track, which get very similar scores. Using the edit distance as a mechanism for tie breaking ensures that *Proceedings* is mapped to *Proceedings* and not to *InProceedings*.

Figure 1 shows the entire process using the introductory example from the OAEI anatomy dataset, computing the similarity score for *tarsal plate* and *eyelid tarsus*.

For multi-lingual ontologies, we first translate the fragments, labels, and comments to English as a pivot language [2], using the Bing Search API's translation capabilities. The translated concepts are then processed as described above. The whole process is illustrated in Fig. 2.



	$\mathrm{doc}_{fragment}$(#MA_0000280)	doc_{label}(#MA_0000280)
$\mathrm{doc}_{fragment}$(#NCI_C33736)	0.04188	0.00844
doc_{label}(#NCI_C33736)	0.02062	**0.21577**

Fig. 1. Example with two concepts from the OAEI anatomy dataset. This is a mono-lingual case; for multi-lingual ontologies, an additional translation step is performed on the extracted search terms.

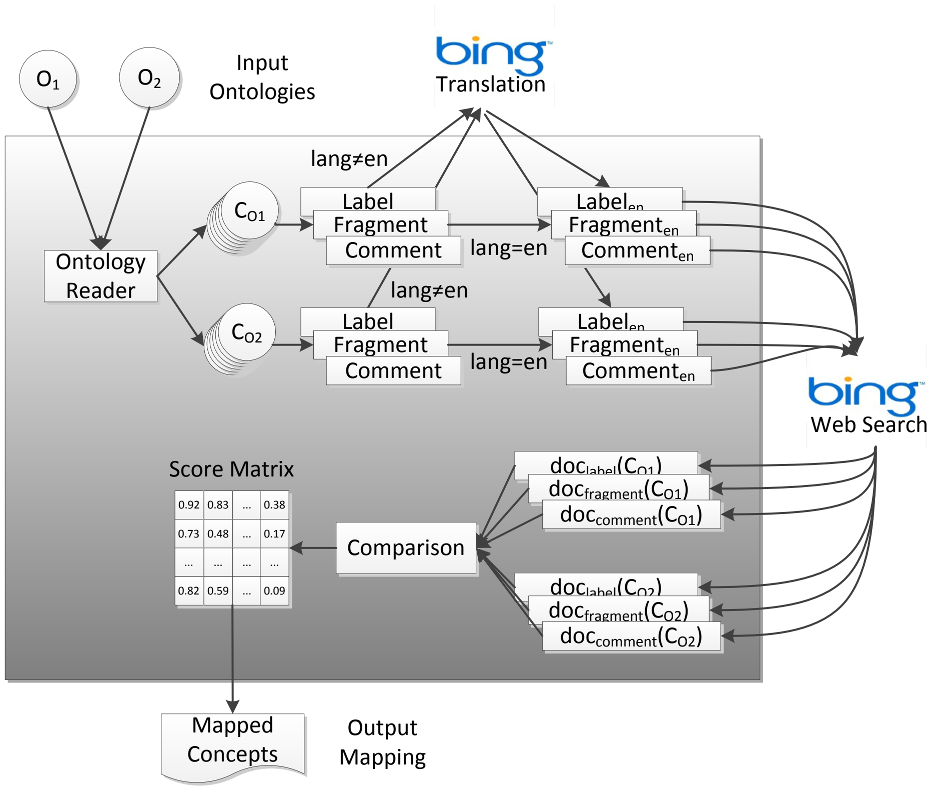


Fig. 2. Illustration of the *WeSeE-Match* matching process. Labels, fragments, and comments are extracted from the input ontologies, translated to English if necessary, and the documents are generated for each concept. A scoring matrix stores the maximum similarities for each pair of concepts. From that matrix, the final mapping is derived.

1.3 Adaptations made for the evaluation

No special adaptations have been made for OAEI 2012. The parameter τ was set to 0.55 for multi-lingual and to 0.6 for mono-lingual matching problems.

1.4 Link to the system and parameters file

The WeSeE-Match tool can be downloaded from `http://www.ke.tu-darmstadt.de/resources/ontology-matching/wesee-match`.

2 Results

2.1 Benchmark

The results on the benchmark set are those expected given the matcher's characteristics. Since the matcher is fully element-based, structural modifications of the ontologies (e.g., removing subclass relations) do not change the results. Furthermore, *WeSeE-Match* relies on natural language identifiers, labels, and comments. Removing those identifiers or replacing them by arbitrary strings creates ontologies where *WeSeE-Match* cannot identify meaningful alignments.

2.2 Anatomy

The results on the anatomy dataset show how background knowledge on the web helps identifying non-trivial mappings. For example, two concepts with the labels *anterior surface of the lens* and *lens anterior epithelium* are matched, one using an English, one a Latin name, as well as two concepts with labels *external ear* and *outer ear*, which are synonyms. As those names are likely to appear on similar web pages, *WeSeE-Match* is capable of identifying them as valid mappings.

2.3 Conference

The results on the conference track show how synonyms (like *Conference Attendee* and *Participant*, or *is reviewing* and *reviewer of paper*) are found by *WeSeE-Match*. The same mechanism, however, sometimes produces false positives of close terms like *Reviewer* and *Member PC*, since those often occur on similar web pages (i.e., conference websites and researchers' CVs).

A general observation is that the performance of *WeSeE-Match* is better with respect to classes than with respect to properties. This can be explained that class labels (such as *author*) make for more concise search terms than relations (such as *written by*).

2.4 Multifarm

Multi-lingual ontologies are well processed by *WeSeE-Match*, resulting in an average F-Measure of 0.41 across all language pairs. The worst results are achieved for Chinese-German (0.24), the best for English-French (0.56), where the latter is close to the performance of *WeSeE-Match* on the mono-lingual conference dataset. As discussed above, *WeSeE-Match* is well capable of identifying mappings between labels that are synonyms. It turns out that the Bing translation service used in *WeSeE-Match* does not provide exact translations, but merely closely related synonyms, such as *camera-ready version of the paper* and *final manuscript*, which are very problematic for string-based processing techniques. As discussed above, *WeSeE-Match* is particularly well suited for matching synonyms. Thus, the combination of translations (which may result in closely related terms) and matching via a web search engine is a good fit.

2.5 Library

Despite its general long run-time (see below), *WeSeE-Match* was capable of completing the larger library track. This track provides many different labels in three languages for most of the concepts, which leads to a lot of search engine requests, but the tool is capable of providing reasonable results at around the same level of quality as on the conference track. This shows that the larger number of labels available in the library track neither helps nor distracts *WeSeE-Match*.

2.6 Large Biomedical Ontologies

Due to a programming error, *WeSeE-Match* was not capable of completing that track.

3 General Comments

3.1 Comments on the Results

The results show that *WeSeE-Match* is capable of producing results that are competitive with state of the art matching tools, despite the very simple approach. Leveraging the knowledge of the world wide web for ontology matching thus appears to be a promising technique. The combination of machine translation and web search appears to be a good fit, because near-exact translations and synonyms are well matched by a search engine based approach.

Being one of the slowest matchers in OAEI, the downside of *WeSeE-Match* clearly is its runtime. However, it is important to notice that *WeSeE-Match* scales *linearly* with respect to runtime. In contrast to approaches such as Normalized Google Distance [1], which require a *quadratic* number of search engine invocations (to compute the number of pages on which a pair of concepts appears together), *WeSeE-Match* creates at most three search engine requests per concepts (one each for the label, the comment, and the URI fragment).

3.2 Possible Improvements of the System

At the moment, *WeSeE-Match* does not make any use of the input ontologies' structure, but is implemented as a purely element-based approach. Possible improvements would include the use of subclass relations as well as domains and ranges of properties. These could, e.g., be included as additional search terms. This could help improving the tool's performance on relations.

Although the tool has only one relevant parameter (the threshold τ), observations have shown that a good choice of this parameter strongly varies among the individual problems. Thus, the choices of this parameter for OAEI 2012 are compromises that provide reasonable, yet not optimal results for all problems. Automatic parameterization techniques [3] could help here in further improving the system's results.

4 Conclusion

The results of *WeSeE-Match* in the OAEI 2012 competition show that an algorithm based on a simple idea – using a standard web search engine and translation service – yields results that can keep up with competitive with tools that have much more complex underlying algorithms.

Given the long run-times, the approach is only applicable in scenarios that do not require real-time results. Furthermore, it is a possible candidate algorithm for dealing with hard-to-solve cases, where the simple cases are solved by faster algorithms. It is rather a candidate to be used in a tool with many matching algorithms to inspect those cases which cannot be handled by simpler algorithms.

References

1. Cilibrasi, R.L., Vitanyi, P.M.: The google similarity distance. IEEE Transactions on Knowledge and Data Engineering **19** (2007) 370–383
2. Paul, M., Yamamoto, H., Sumita, E., Nakamura, S.: On the importance of pivot language selection for statistical machine translation. In: 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers. (2009) 221–224
3. Ritze, D., Paulheim, H.: Towards an automatic parameterization of ontology matching tools based on example mappings. In: Proceedings of the 6th International Workshop on Ontology Matching, Bonn, Germany, October 24, 2011. Volume 814 of CEUR WS. (2011)