WikiMatch Results for OEAI 2012

Sven Hertling and Heiko Paulheim

Technische Universität Darmstadt
{hertling,paulheim}@ke.tu-darmstadt.de

Abstract. WikiMatch is a matching tool which makes use of Wikipedia as an external knowledge resource. The overall idea is to search Wikipedia for a given concept and retrieve all pages describing the term. If there is a large amount of common pages for two terms, then the concepts will have similar semantics. We make also use of the inter-language links between Wikipedias in different languages to match multilingual ontologies. The results show that this simple idea can keep up with state of the art tools. Moreover, the results on the Multifarm track depend on the Wikipedia's number of articles as well as the link amount to the Wikipedia of the other natural language to match. The growth of Wikipedia will thus help this matcher to improve the matching quality.

1 Presentation of the system

1.1 State, purpose, general statement

WikiMatch is an element-level ontology matching tool. It uses Wikipedia as a huge background knowledge to find out, how similar two concepts are. The algorithm extracts all labels, comments, and URI fragments, and uses Wikipedia's search function to retrieve an set of articles related to that term. If the intersection between such two sets is high, then we assume that the terms have something in common and are related to each other.

To also deal with multilingual ontologies, all language links of the returned articles are requested as a second step. For each language, the Jaccard coefficient of the two sets of articles retrieved is computed, as equation (1) shows.

$$sim(t_1, t_2) := max_{t_i \in \{label(c_i), fragment(c_i), comment(c_i)\}, i \in \{1,2\}} \frac{\#(S(t_1) \cap S(t_2))}{\#(S(t_1) \cup S(t_2))} \tag{1}$$

For the terms $t1$ and $t2$ the resulting article set S is computed. The maximum over all labels, comments and URI fragments are then the similarity measure for these terms.

If Wikipedia returns an suggestion for the term, a new query is made with this new search term. This is typically the case when entering a misspelled term in the search. An overview of the entire system is shown in Fig. 1.

1.2 Specific techniques used

Our first test was to search for the whole term in Wikipedia. We call this approach *simple search*. As a result the precision is high in contrast to the recall which is very low. To

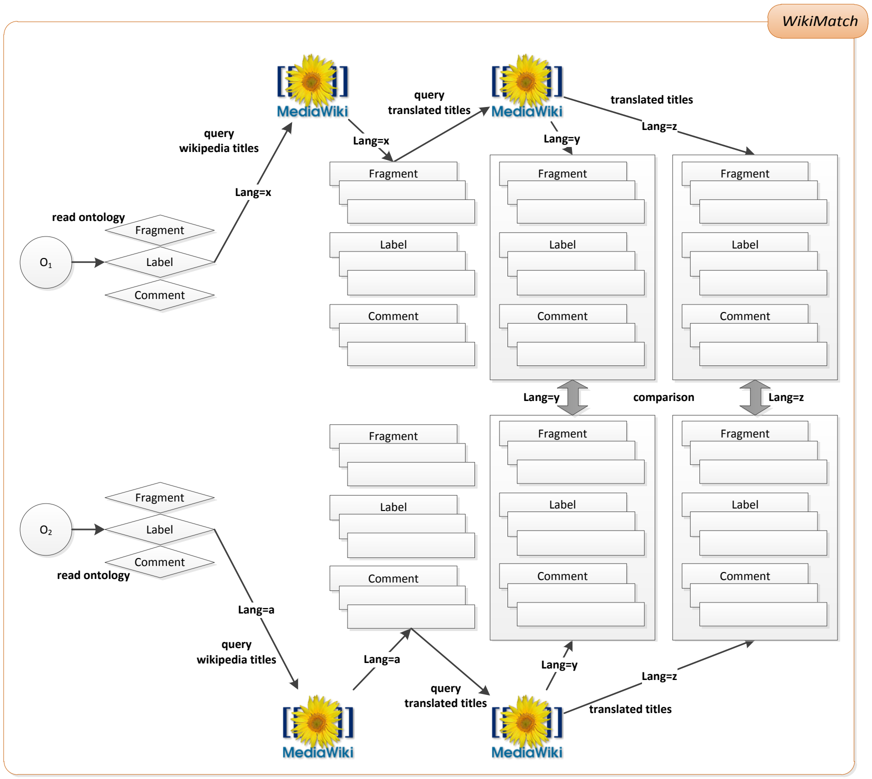

Fig. 1. Illustration of the matching process (see [1]). As a first all Wikipedia articles are requested for the language of the term. As a second step all language links from these articles are queried. The comparison of all these sets is per language. The maximum of the cross product of fragment, comment and label is returned.

improve the recall measure we have tried another search approach, i.e., splitting each term into individual tokens and searching for those tokens individually. For example, the query for the string *Passive_conference_participant* will therefore contain three single searches with $passive$, $conference$ and $participant$. Both search approaches are shown in Fig. 2 in pseudo code.

Our own tests showed that the *individual tokens search* (ITS) will result in a better recall, but a lower precision. To have a look at the F-Measure between the two approaches, the first idea of *simple search* can produce better values. Therefore this approach is was submitted.

1.3 Adaptations made for the evaluation

No adaptions for the evaluation are made.

```
float getsimilarity(term1, term2) {
  titlesForTerm1 = getAllTitles(term1);
  titlesForTerm2 = getAllTitles(term2);

  commonTitles = intersectionOf(titlesForTerm1, titlesForTerm2);
  allTitles = unionOf(titlesForTerm1, titlesForTerm2);

  return #(commonTitles) / #(allTitles);
}

List<WikipediaPage> getAllTitles(searchTerm) {
  removeStopwords(searchTerm);
  removePunctuation(searchTerm);

  if(simpleSearch) {
    resultList = searchWikipedia(searchTerm);
  }

  if(individualTokenSearch) {
    tokens = tokenize(searchTerm);
    for each token in tokens
      resultList = resultList + searchWikipedia(searchTerm);
  }

  for each page in results
    resultList = resultList + getLanguageLinks(page);

  return resultList;
}
```

Fig. 2. Pseudo code of *simple search* and *individual token search* (see [1]).

1.4 Link to the system and parameters file

The WikiMatch tool can be downloaded from `http://www.ke.tu-darmstadt.de/resources/ontology-matching/wikimatch`.

2 Results

2.1 Benchmark

Since our approach is entirely element-based, removing or replacing labels or comments results in lower F-Mesaure. By removing only one of the describing elements, WikiMatch deals also with the remaining literals and can provide good results. If there are neither labels nor comments, then this approach does not work. On the other hand, removing structural features, such as subclass relations, does not influence the results of WikiMatch.

2.2 Anatomy

The comparison with *StringEquiv* of the OAEI 2011.5 is not that well, because the recall is not much higher, but therefore the precision is very low (0.997 to 0.864). A nontrivial mapping that is found by our tool is *ophthalmic artery* and *Opthalmic_Artery*.

2.3 Conference

In the conference track, WikiMatch reached 0.6 F-Measure for ra1. This is better than the baseline2 from OAEI 2011.5. The same applies for ra2. Unfortunately, the conference domain is not well covered in Wikipedia to match special terms like *Chair_PC* and *ProgramCommitteeChair*. But through the suggestion feature it is possible to find a mapping between *Sponsorship* and *Sponzorship*.

2.4 Multifarm

On the Multifarm track, WikiMatch exploits the inter-language links from each returned article. Therefore a mapping between different languages can be found. The best results are achieved for matching English to Spanish (F-Measure 0.29), the worst for Chinese-German and Chinese-Portuguese (F-Measure 0.1).

The results on the Multifarm track strongly depend on the involved Wikipedia's sizes, in particular the number of articles and links to other Wikipedias. Fig. 3 depicts the results of WikiMatch in relation to the corresponding Wikipedias' article counts; Fig. 4 the results in relation to the number of links from the corresponding Wikipedias to other Wikipedias[1]. It can be observed that the results get better with larger and more strongly inter-linked Wikipedias.

As the number of articles and inter-Wikipedia links grow by around 2% per month (even more rapidly for Chinese, which is currently the smallest and least interlinked Wikipedia used in Multifarm), we expect the results of WikiMatch to improve just by the growth of Wikipedia. The trend lines in Fig. 3 and 4 indicate that about 500,000 additional articles and Wikipedia links lead to an increase of five percentage points in F-Measure. At the current growth rate of Wikipedia, this takes a little less than two years.

2.5 Library

The library track unfortunately did not finish within one week. The reason can be the calculation of the cross product between the concepts of the ontologies, or the generally long times required for looking up concepts in Wikipedia. This requires an more detailed look.

2.6 Large Biomedical Ontologies

Like the library track, the ontologies in this track are also too large handle by WikiMatch in its current version.

[1] Using numbers obtained from `http://stats.wikimedia.org/`

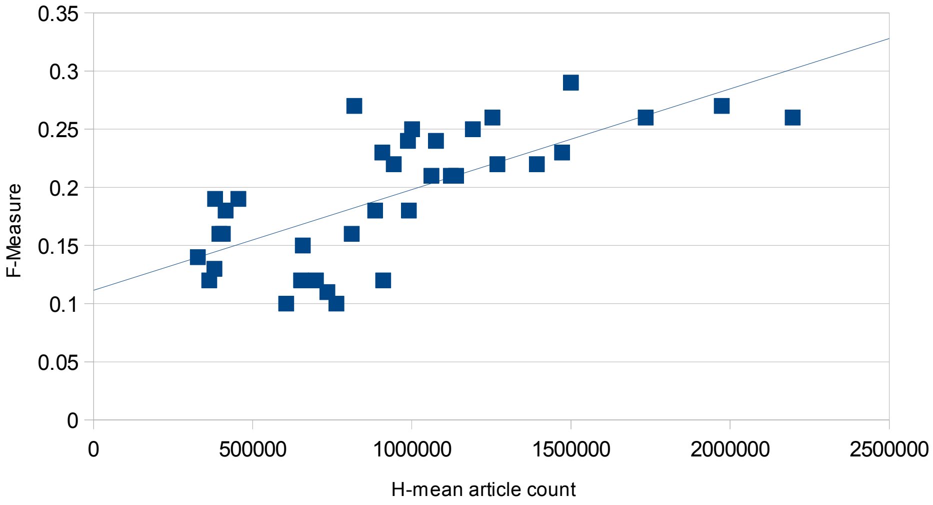

Fig. 3. Multifarm results in relation to the corresponding Wikipedias' article counts

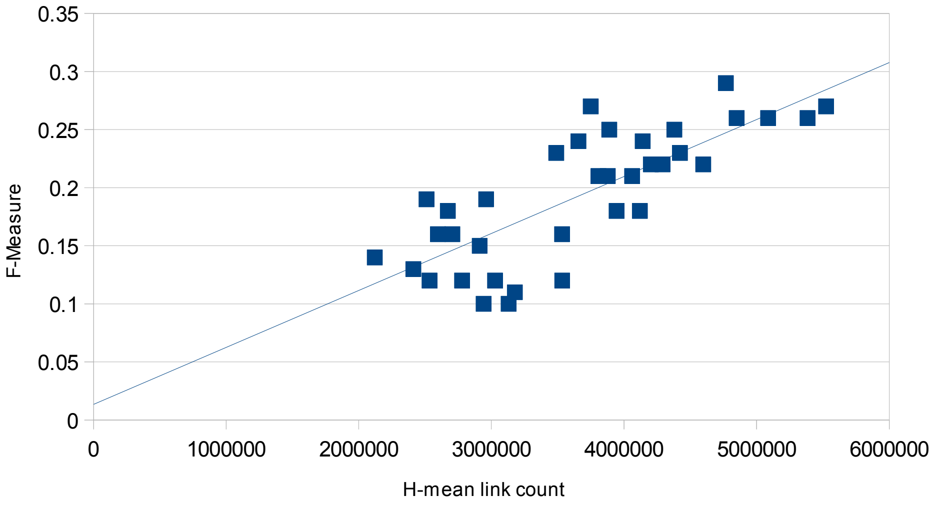

Fig. 4. Multifarm results in relation to the corresponding Wikipedias' inter-wiki link counts

3 General comments

3.1 Comments on the results

On Multifarm and conference track WikiMatch shows that a simple element based approach can keep up with state of the art tools. Especially using the inter-language links in Wikipedia looks like a promising approach to deal with multi-lingual ontologies. On large tracks the current approach does not scale well and did not finish in time.

In general, like most approaches using web data by querying the web at run-time, WikiMatch is rather slow compared to matchers working entirely internally or only use local resources.

3.2 Discussions on the way to improve the proposed system

For improving the approach, we envision to set threshold values dynamically, based on the matched ontologies. In order to cope with the run-time restrictions, it is possible to not use WikiMatch as a single matching approach, but to first match the easy cases (i.e., same or very similar terms) with string-level methods.

At the moment, WikiMatch only uses the page identifiers returned by the search, ignoring the text snippets, i.e., the portions of the Wikipedia pages that are relevant for the search term. Using those snippets, e.g., like WeSeE-Match does [2], could help leveraging the potential of WikiMatch more effectively.

4 Conclusion

With our work on WikiMatch, we have shown how a large general-purpose resource like Wikipedia can be used for ontology matching. Especially the cross-linking of different language Wikipedias is useful for multi-lingual ontology matching. Furthermore, we have seen that the results of WikiMatch improve with a growing size of Wikipedia – which in turn indicates that the results of WikiMatch will improve in the future merely by the growth of Wikipedia.

References

1. Hertling, S., Paulheim, H.: Wikimatch - using wikipedia for ontology matching. In: Seventh International Workshop on Ontology Matching (OM 2012). (2012)
2. Paulheim, H.: Wesee-match results for oeai 2012. In: Seventh International Workshop on Ontology Matching (OM 2012). (2012)