StringsAuto and MapSSS Results for OAEI 2013

Michelle Cheatham and Pascal Hitzler

Kno.e.sis Center, Wright State University, Dayton, OH, USA
{cheatham.7, pascal.hitzler}@wright.edu

Abstract. StringsAuto and MapSSS are two closely related ontology alignment systems. The StringsAuto matcher seeks to explore the limits of a syntactic-only approach to alignment. The MapSSS system then expands on this work by embedding the syntactic matching of StringsAuto within a more complete alignment system that also makes use of semantic and structural information. In this paper we describe the basic operation of the two systems and discuss their performance in the OAEI 2013 evaluation.

1 Presentation of the system

1.1 State, purpose, general statement

The vast majority of ontology alignment systems use some form of string similarity metric. Our overall goal with StringsAuto and MapSSS is to explore the importance of the choice of a particular string metric. StringsAuto consists *only* of string metrics, while MapSSS uses strategically chosen string metrics within the context of a more fully-featured alignment system.

In [1] we analyzed the performance of eleven string similarity metrics (TF-IDF, Soft TF-IDF, Jaccard, Soft Jaccard, Exact Match, Longest Common Substring, Jaro Winkler, Levenstein, Monge Elkan, N-gram, and Stoilos) on different types of ontologies (standard, biomedical, and multi-lingual). In addition, we experimented with the use of common string pre-processing methods (tokenization, normalization, stemming, stop word removal, synonyms, and translations). StringsAuto grew out of this work. Its purpose is to investigate string similarity metrics as applied to ontology alignment. In particular, it is of interest to compare the performance of this system to that of the very basic string-based matchers used as baselines for some of the OAEI tracks.

The MapSSS system was involved in previous OAEI evaluations. The three S's in MapSSS stand for syntactic, semantic, and structural, which are the three types of metrics used by the system. This year MapSSS has been augmented with a different semantic metric, based on Google queries, and modified to use the same syntactic metric selection strategy as StringsAuto. We are interested in comparing the performance of this version to that of previous years.

1.2 Specific techniques used

Based on the results of the string metric analysis in [1], we produced a set of guidelines for choosing string metrics and preprocessing strategies based on the characteristics of

the ontologies to be aligned and whether precision or recall is of primary concern. More information can be found in the referenced paper.

- Precision
  - Less than two words per label: **Jaro-Winkler 1, 1**
  - Two or more words per label
    - Synonyms: **Soft Jaccard .2, .5 with Levenstein .9 base metric**
    - No synonyms: **Soft Jaccard 1, 1 with Levenstein .8 base metric**
- Recall
  - Less than two words per label: **TF-IDF .8, .8**
  - Two or more words per label
    - Synonyms: **Soft TF-IDF .5, .8 with Jaro-Winkler .8 base metric**
    - Different Languages: **Soft TF-IDF 0, .7 with Jaro-Winkler .9 base metric**
    - Other: **Soft TF-IDF .8, .8 with Jaro-Winkler .8 base metric**

StringsAuto simply chooses two metrics based on these heuristics: one that prioritizes precision and another that focuses on recall. Each of these metrics is run (in series) and the resulting alignment is used as-is. When a metric is run, every label in the first ontology is compared to every label in the second ontology, and the results of the similarity metric are stored in a matrix. The stable marriage algorithm is then run over the matrix, and any matches greater than a threshold value are included in the alignment. If either entity involved in a match has already been used in the alignment, that match is ignored. This means that all alignments generated are 1:1 and the recall-centric metric cannot override the precision-centric method.

MapSSS uses the same syntactic metric selection strategy as StringsAuto. In addition, it uses a semantic metric based on Google queries. When considering two labels, A from the first ontology and B from the second, this metric queries Google for the phrase A definition. It then searches the snippets on the first page of results for B. If B is found, the metric returns true, otherwise it returns false. If this metric returns true in both directions (i.e. googleMetric(A, B) and googleMetric(B, A) are both true) then the mapping is added to the alignment. Finally, MapSSS also contains a structural metric. If all of the entities in the direct neighborhood of two classes are mapped to one another, then those classes are mapped. This approach is sometimes called "flooding." The structural metric is run repeatedly until no new mappings are created.

1.3 Adaptations made for the evaluation

No significant adaptations were made for the OAEI evaluation. In particular, the heuristics used to select the string similarity metrics do not break cleanly along the different OAEI tracks. Some possibly relevant details of these alignment systems include:

- Neither alignment system attempts to align properties or instances; only classes are considered. Our previous work has shown that string similarity metrics perform particularly poorly on property labels.

– The systems determine the language of an ontology by randomly selecting a sample of ten entity labels and sending them to Google Translate. This assumes each ontology involves predominately one language.
– To determine if an ontology has embedded synonyms, the alignment systems look for tags involving the word "synonym." This is to some extent tailored to the anatomy track of the OAEI.
– The semantic metric within MapSSS uses the Google API. There is a limit on the number of queries that can be submitted using this API each day, as well as a monthly cap. This causes problems for some of the larger ontology alignment problems within the OAEI evaluation. We attempted to cache the query results to alleviate this problem, but the SEALS server configuration made this unworkable (we would need to be able to write to a file during execution and have this file available during subsequent runs of the program).

1.4 Link to the system and parameters file

StringsAuto is available at http://pascal-hitzler.de/resources/Strings.zip and MapSSS is available at http://pascal-hitzler.de/resources/MapSSS.zip.

2 Results

Development and testing of StringsAuto and MapSSS focused primarily on the conference, anatomy, and multiform test sets, but we present results for all tracks in which alignments were produced.

2.1 anatomy

StringsAuto achieved an f-measure of 0.835 on this test set (see Table 1). This placed it 7th out of 21 participating systems. In particular, the results produced by StringsAuto were significantly better than those of StringsEquiv, a basic string equality matcher.

Interestingly, the performance of MapSSS did not differ greatly from StringsAuto. When compared to the performance of the 2012 version of MapSSS, we see that the precision has dropped while the recall has increased slightly. Notably, the recall+ measure is significantly higher with the current version, which makes use of a string similarity metric specifically chosen to enhance recall and semantic information gleaned from Google queries.

2.2 conference

The (ra2) results of StringsAuto and MapSSS on the conference track are shown in Table 2. StringsAuto outperformed both StringsEquiv and edna (an edit distance metric with a threshold of .82). Overall, StringsAuto was 6th out of 27 alignment systems in terms of f-measure, while edna was 11th and StringsEquiv was 22nd. The 2013 version of MapSSS significantly outperformed its predecessor but fell slightly short of StringsAuto.

Table 1. Anatomy Track Results

Alignment System	F-measure	Precision	Recall	Recall+
StringsEquiv	.766	.997	.622	0
StringsAuto	.835	.899	.779	.433
MapSSS 2013	.828	.898	.768	.443
MapSSS 2012	.831	.935	.747	.337

Table 2. Conference Track Results

Alignment System	F-measure	Precision	Recall
StringsEquiv	.52	.76	.39
edna	.55	.73	.44
StringsAuto	.60	.74	.50
MapSSS 2013	.58	.77	.46
MapSSS 2012	.46	.47	.46

2.3 multifarm

There was a problem running both StringsAuto and MapSSS on the multifarm test set. While both systems were able to produce alignments, they had to fall back to their non-translating versions due to a problem reaching the Google Translate service from the OAEI test server. We attempted to fix this during the evaluation by caching the results of the translation queries, but this did not work, possibly due to write restrictions on the server itself (we need to be able to write to a file that will persist between different executions of the program). Here we report both the results achieved during the evaluation and the results we get when we run StringsAuto on a local computer. In addition, the code used to generate the StringsAuto results is available from http://pascal-hitzler.de/resources/Strings.zip and the actual alignments produced on the multifarm test cases are available http://pascal-hitzler.de/resources/Multifarm2013alignments.zip.

Table 3. Multifarm Track Results

	Different			Same		
Alignment System	F-measure	Precision	Recall	F-measure	Precision	Recall
StringsAuto	.14	.30	.09	.07	.51	.04
MapSSS	.10	.27	.07	.06	.50	.03
StringsAuto (corrected)	.30	.42	.23	.36	.92	.23

2.4 library

MapSSS did not produce alignments for this track, likely due to the size of the thesauri causing the system to exceed the Google API query limit.

StringsAuto finished below the reference (string equality) matchers on this test. StringsAuto could very probably be improved by recognizing all of the labels as synonyms (as it does for the anatomy benchmark). Another potential issue is that StringsAuto might have decided that either or both of the ontologies was entirely in German due to its sampling technique, and then attempted to translate all of the labels in that ontology (even the ones already in English).

Table 4. Library Track Results

Alignment System	F-measure	Precision	Recall
StringsAuto	.302	.774	.188

2.5 large biomedical ontologies

In this track StringsAuto was only able to complete one out of the six tasks (FMA-NCI), and MapSSS was not able to complete any of them (again due to the Google API query limit).

The results of StringsAuto on the FMA-NCI task were not very good. The system achieved an f-measure of 0.667 (based on a precision of 0.838 and a recall of 0.554). This placed the system 20th out of 23. It is odd that the performance here is so different than on the anatomy track. Similar to the problem on the library track, it is likely this is partially due to StringAuto's inability to recognize multiple labels for a single entity as synonyms. The synonym extraction method should be adapted to include this information.

It might be surprising that StringsAuto was unable to complete more of the tasks in this track. While in theory a matcher that only does string comparisons of labels should scale very well, StringsAuto uses a global (m x n) matrix to store all of the pair-wise similarity values and runs the stable marriage algorithm over this data. This obviously runs into memory limitations for large ontologies. In the future it might make sense to choose mappings based on a simple local maximum (with a threshold).

3 General comments

3.1 Comments on the results

Despite some technical problems, the performance of StringsAuto compared to that of the base string matchers shows that a careful selection of string similarity metrics leads to a significant performance increase in ontology alignment systems. In fact, StringsAuto finished in the top third of all alignment systems in both the anatomy and conference tracks. This shows that a significant amount of semantic information within some ontologies is contained in the labels themselves, and string similarity metrics are therefore an important component of ontology alignment systems.

The lackluster performance of MapSSS when compared with StringsAuto was somewhat surprising. Further research will be needed to improve the utility of the Google-based semantic similarity metric. We have begun looking into leveraging other general sources of information, including wikilinks[1] and Wikipedia. It would be interesting to perform an comprehensive analysis of these type of metrics similar to the one done for string similarity metrics in [1].

3.2 Discussions on the way to improve the proposed system

While StringsAuto is basically a proof-of-concept alignment system, it could be extended in several ways that would improve its performance on the OAEI evaluation. In particular, it could be adapted to treat multiple labels for a single entity as synonyms and to avoid the use of a global data structure so that larger ontology pairs could be aligned.

The main problem with MapSSS is due to the Google API query limit. This is also a problem with Bing, according to their terms of service. To mitigate this issue, we need to identify another general information source that does not have such a limit or only invoke this metric in a more limited way.

3.3 Comments on the OAEI 2013 procedure

It would be convenient to provide a way to run all of the language pairs in the multifarm test set with a single command and produce the same results published by the organizers of that track (i.e. the precision, recall and f-measure separated into the "same" and "different" ontology categories).

3.4 Proposed new measures

It might be interesting to see some details about the alignments produced by the various tools. For instance, were there some mappings identified by all of the alignment systems? Were there some that were missed by all systems? This might provide insights that improve the performance of ontology alignment systems in general. It might also highlight any controversial mappings remaining in the reference alignments.

4 Conclusion

We have described two related ontology alignment systems, StringsAuto and MapSSS, which explore the role that string similarity metrics play in ontology alignments. The results of these matchers on the OAEI evaluation are significantly better than the baseline string similarity matchers, and in some cases perform quite well when compared to all other alignment systems. The disappointing performance of the Google-based semantic similarity metric used in MapSSS indicates the need for further research in this area.

[1] http://www.iesl.cs.umass.edu/data/wiki-links

Acknowledgements

This work was supported by the National Science Foundation under award 1354778 "EAGER: Collaborative Research: EarthCube Building Blocks, Leveraging Semantics and Linked Data for Geoscience Data Sharing and Discovery." Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. M. Cheatham and P. Hitzler. String similarity metrics for ontology alignment. In *Proceedings of the 12th International Semantic Web Conference (ISWC 2013), Sydney, NSW, Australia, October 21-25, 2013*, Heidelberg, 2013. Springer.