AML Results for OAEI 2015

Daniel Faria[1], Catarina Martins[2], Amruta Nanavaty[4],
Daniela Oliveira[2], Booma Sowkarthiga[4], Aynaz Taheri[4],
Catia Pesquita[2,3], Francisco M. Couto[2,3], and Isabel F. Cruz[4]

[1] Instituto Gulbenkian de Ciência, Portugal
[2] Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, Portugal
[3] LASIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal
[4] ADVIS Lab, Department of Computer Science, University of Illinois at Chicago, USA

Abstract. AgreementMakerLight (AML) is an automated ontology matching
system based primarily on element-level matching and the use of external re-
sources as background knowledge. This paper describes its configuration for the
OAEI 2015 competition and discusses its results.
This year, we focused mainly on the Interactive Matching track due to its expan-
sion, as handling user interactions on large-scale tasks is a critical challenge in
ontology matching.
AML's participation in the OAEI 2015 was very successful, as it improved upon
last year's results in 4 of the 7 ontology matching tracks.

1 Presentation of the system

1.1 State, purpose, general statement

AgreementMakerLight (AML) is an automated ontology matching system based pri-
marily on lexical matching techniques, with an emphasis on the use of external re-
sources as background knowledge and on alignment coherence. While originally fo-
cused on the biomedical domain, AML's scope has been expanded, and it can now be
considered a general-purpose ontoloy matching system.
AML was derived from AgreementMaker [1, 2] and combines its design principles
(flexibility and extensibility) with a strong focus on efficiency [5]. It draws on the
knowledge accumulated in AgreementMaker by reusing and adapting several of its
components, but also includes a number of novel components.
This year, our development of AML for the OAEI competition focused primarily on
the Interactive Matching track, due to its expansion to include the Anatomy and Large
Biomedical Ontologies datasets. Handling user feedback on large-scale tasks is a criti-
cal challenge in ontology matching, and we felt that tackling this new challenge should
be our main goal.

1.2 Specific techniques used

The AML workflow for the OAEI 2015 is the same as last year, comprising the nine
steps shown in Figure 1: ontology loading and profiling, translation, baseline match-

ing, background knowledge matching, word and string matching, structural matching , property matching, selection, and repair.
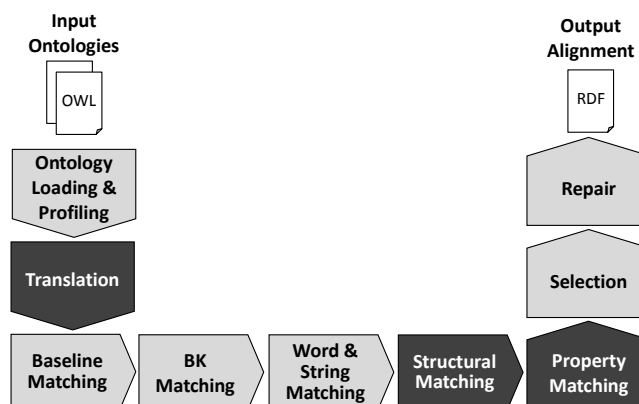

Fig. 1. The AgreementMakerLight matching workflow for the OAEI 2015. Steps in dark gray are conditional.

Ontology Loading & Profiling AML employs the OWL API [6] to read the input ontologies then retrieve the necessary information to populate its own data structures [5]:

- Class local names, labels and synonym annotations are normalized and stored into the *Lexicon* of the corresponding ontology. AML automatically derives new synonyms for each name by removing leading and trailing stop words [10], and by removing name sections within parenthesis.
- Property local names and labels are also stored in the Lexicon, while their types, domains, and ranges are stored in the *Ontology* as *Property* objects.
- Relations between classes (including disjointness) and between properties are stored in a global *RelationshipMap*.

AML does not store or use comments, definitions, or instances.
After loading, the matching problem is profiled taking into account the size of the ontologies, their language(s), and the property/class ratio.

Translation AML features an automatic translation module based on Microsoft® Translator. When there is no significant overlap between the language(s) of the input ontologies, AML employs this module to translate the names of all classes and properties from the language(s) of the first ontology to the language(s) of the second and vice-versa. The translation is done by querying Microsoft® Translator for the full name (rather than word-by-word). To improve performance, AML stores locally all translation results in dictionary files, and queries the Translator only when no stored translation is found.

Baseline Matching AML employs an efficient weighted string-equivalence algorithm, the *Lexical Matcher* [5], to obtain a baseline class alignment between the input ontologies. The *Lexical Matcher* has been updated to handle multi-language ontologies, by matching only class names in the same language.

Background Knowledge Matching AML has available four sources of background knowledge which can be used as mediators between the input ontologies: the Uber Anatomy Ontology (Uberon) [8], the Human Disease Ontology (DOID) [12], the Medical Subject Headings (MeSH) [9], and the WordNet [7].
The WordNet is only used for small English language ontologies, as it is prone to produce erroneous mappings in other settings. It is used through the JAWS API [1] and with the *Lexical Matcher*. The remaining three background knowledge sources are tested in all non-small single-language problems, by measuring their mapping gain over the baseline alignment [4]. When their mapping gain is very high ($\geq 20\%$), they are used to extend the *Lexicons* of the input ontologies [10]; otherwise, when it is above the minimum threshold (2%) their alignment is merged with the baseline alignment.
Uberon and DOID are both used in OWL format, and each has an additional table of pre-processed cross-references (in a text file). They can be used directly through the cross-references or with the *Lexical Matcher*. MeSH is used as a stored *Lexicon* file, which was produced by parsing the MeSH XML file, and is used only with the *Lexical Matcher*.

Word & String Matching To further extend the alignment, AML employs a word-based similarity algorithm (the *Word Matcher*) and a string similarity algorithm (the *Parametric String Matcher*) [5]. The former is not used for very large ontologies, because it is error prone. The latter is used globally for small ontologies, but only locally for larger ones as it is time-consuming.
For small ontologies, AML also employs the *Multi-Word Matcher*, which matches closely related multi-word names that have matching words and/or words with common WordNet synonyms or close hypernyms, and the new *Acronym Matcher*, which attempts to match acronyms to the corresponding full name.

Structural Matching For small and medium-sized ontologies, AML also employs a structural matching algorithm, called *Neighbor Similarity Matcher*, that is analogous to AgreementMaker's Descendants Similarity Inheritance algorithm [3]. This algorithm computes similarity between two classes by propagating the similarity of their matched ancestors and descendants, using a weighting factor to account for distance.

Property Matching When the input ontologies have a high property/class ratio, AML also employs the *PropertyMatcher*. This algorithm first ensures that properties have the same type and corresponding/matching domains and ranges. If they do, it compares the properties' names by doing a full-name match and computing word similarity, string similarity, and WordNet similarity.

---

[1] http://lyle.smu.edu/ tspell/jaws/

Selection AML employs a greedy selection algorithm, the *Ranked Selector* [5], to reduce the cardinality of the alignment. Depending on the size of the input ontologies, one of three selection strategies is used: strict, permissive, or hybrid. In strict selection, no concurrent mappings (i.e., different mappings for the same class/property) are allowed and a strict 1-to-1 alignment is produced; in permissive selection, concurrent mappings are allowed if their similarity score is exactly the same; in hybrid selection, up to two mappings per class are allowed above 75% similarity, and permissive selection is applied below this threshold. For very large ontologies, AML employs a selection variant that consists on combining the (lexical) similarity between the classes with their structural similarity, prior to performing ranked selection. This strategy enables AML to select mappings that "fit in" structurally over those that are outliers but have a high lexical similarity.

In interactive matching mode, AML employs an interactive selection algorithm instead. This algorithm uses patterns in the similarity values produced by AML's various matching algorithms to detect suspicious mappings. Above the high similarity threshold of 70% AML queries the user for suspicious mappings, and accepts all other mappings as true. Below this threshold, AML automatically rejects suspicious mappings, and queries the user for all other mappings, until the minimum threshold of 45% is reached, the limit of consecutive negative answers is reached, or the query limit is reached, whichever happens first. The query limit is 45% of the alignment for small ontologies, and 15% of the alignment for all other ontologies (with a further 5% of the alignment reserved for interactive repair), ensures that the workload for the user is kept within reasonable boundaries.

Repair AML employs a heuristic repair algorithm to ensure that the final alignment is coherent [11].

For the interactive matching track, AML employs an interactive variant of this algorithm, wherein the user is asked for feedback about the mappings selected for removal. This variant is not used on the Large Biomedical Ontologies dataset due to its particular evaluation, wherein mappings repaired from the reference alignment are ignored but considered true by the Oracle.

1.3 Adaptations made for the evaluation

The only adaptations made for the evaluation were the preprocessing of cross-references from Uberon and DOID for use in the Anatomy and Large Biomedical Ontologies tracks (due to namespace differences), and the precomputing of translations for the Multifarm track (due to Microsoft® Translator's query limit).

1.4 Link to the system and parameters file

AML is an open source ontology matching system and is available through GitHub (https://github.com/AgreementMakerLight).

1.5 Link to the set of provided alignments

The alignments generated by AML for the OAEI 2015 are available at the SOMER project page (http://somer.fc.ul.pt/).

2 Results

2.1 Anatomy

AML had almost identical results to last year, with an F-measure of 94% and a recall++ of 82%, making it the best performing system in this track this year as well. The only difference from last year's alignment was one missing mapping due to a change in the structural matching algorithm.

2.2 Benchmark

N/A.

2.3 Conference

AML had the best performance overall in the Conference track, with the highest F-measure on the full reference alignments ra1 and ra2 (74% and 70% respectively). It also had the highest F-measure in the class-only alignments, and the second-highest in the property-only alignments (notably with 100% precision). In comparison with last year, AML improved its F-measure by 3% with regard to ra2, thanks to the addition of the *Acronym Matcher* and to a few refinements in the processing of properties. Concerning the logical reasoning evaluation, AML was one of the five systems that produced alignments without consistency principle violations, and it had an average number of conservativity principle violations of 1.86 which is the sixth lowest overall, and a reasonable figure considering some of these violations are false positives.

2.4 Interactive Matching

AML obtained the highest F-measure in the Anatomy and Conference datasets, with 96.2% and 81.8% respectively (with no error). Furthermore, it made the lowest number of queries to the Oracle in Conference, and the second lowest in Anatomy. As for the LargeBio datasets, AML had an average F-measure of 84.5% while making an average of 1581 queries to the Oracle.

In comparison with its non-interactive results, AML improved between 1.8-3% in F-measure in the Anatomy and LargeBio tasks and nearly 8% in the Conference task, generally improving both precision and recall (with the sole exception being the SNOMED-NCI small fragments task).

With regard to the error, as would be expected, AML's performance decreased as the error increased, and at an error rate of 30% its performance was always worse than the non-interactive one. Furthermore, because AML increases both precision and recall in interactive mode, the error naturally affected both parameters.

2.5 Large Biomedical Ontologies

AML obtained the same results as last year in the LargeBio track, with F-measures of 92.8% and 84.4% in FMA-NCI (small fragments and whole ontologies respectively), 82.4% and 75.1% in FMA-SNOMED, and 80.9% and 75.6% in SNOMED-NCI. This was expected as no new developments were made to AML with regard to this track.

2.6 Multifarm

AML obtained an F-measure of 52.4% in the different ontologies modality, and of 65.8% in the same ontologies modality. Since these results are not directly comparable to last year, due to the introduction of the Arabic language ontologies, we also tested AML on last year's dataset. It obtained an F-measure of 54.4% in the different ontologies modality (0.1% improvement over last year), and of 67.1% in the same ontologies modality (3.3% improvement). These improvements are due to the refinement of the *Neighbor Similarity Matcher* algorithm, which is naturally more effective when matching the same ontology (albeit in different languages) given that there will be a perfect structural alignment in that case.

2.7 Ontology Alignment for Query Answering

AML had the best performance in this track this year, with an F-measure of 75.9% using the original reference alignment (ra1) and 74.4% using the repaired reference alignment (rar1). It also had the highest precision (tied with XMap on ra1) and recall (tied with LogMap on both ra1 and rar1). These results reflect the fact that AML was the best performing system in the Conference track.

3 General comments

3.1 Comments on the results

In comparison with last year, AML improved its performance in 4 tracks: Conference, Interactive Matching, Multifarm, and Ontology Alignment for Query Answering. Its performance in the Anatomy and LargeBio tracks was essentially the same as last year.

3.2 Discussions on the way to improve the proposed system

The main feature missing from AML is handling and matching instances, and thus the one which would allow it to improve the most.

3.3 Comments on the OAEI test cases

The expansion of the Interactive Matching track to include more challenging test cases and simulate user error was a welcome improvement to this track and to the OAEI as a whole. Alas, not all was perfect with this year's evaluation, as the Oracle's behaviour

on the LargeBio 'soft' repaired reference alignments severely hindered the performance of any interactive repair algorithm, and led to our opting not to employ ours on the LargeBio datasets. We also believe that a query limit should be enforced to ensure that the usage of the Oracle remains within reasonable boundaries, and that systems do not employ the Oracle to review all their mapping candidates.

4   Conclusion

AML built upon its success in last year's evaluation by improving its interactive selection algorithm and refining its strategy for matching small ontologies. This has led to performance improvements in the Interactive Matching track as well as in all tracks based on the Conference dataset. AML's performance in the Anatomy and LargeBio tracks was essentially the same as last year, as we felt that investing further in these tracks would bring diminishing returns given that AML's performance was already very good. Thus, we made no significant changes to the code affecting these tracks.

Acknowledgments

References

1. I. F. Cruz, F. Palandri Antonelli, and C. Stroe. AgreementMaker: Efficient Matching for Large Real-World Schemas and Ontologies. *PVLDB*, 2(2):1586–1589, 2009.
2. I. F. Cruz, C. Stroe, F. Caimi, A. Fabiani, C. Pesquita, F. M. Couto, and M. Palmonari. Using AgreementMaker to Align Ontologies for OAEI 2011. In *ISWC International Workshop on Ontology Matching (OM)*, volume 814 of *CEUR Workshop Proceedings*, pages 114–121, 2011.
3. I. F. Cruz and W. Sunna. Structural alignment methods with applications to geospatial ontologies. *Transactions in GIS*, 12(6):683–711, 2008.
4. D. Faria, C. Pesquita, E. Santos, I. F. Cruz, and F. M. Couto. Automatic Background Knowledge Selection for Matching Biomedical Ontologies. *PLoS One*, In Press, 2014.
5. D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto. The AgreementMakerLight Ontology Matching System. In *OTM Conferences - ODBASE*, pages 527–541, 2013.
6. M. Horridge and S. Bechhofer. The owl api: A java api for owl ontologies. *Semantic Web*, 2(1):11–21, 2011.
7. G. A. Miller. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41, 1995.
8. C. J. Mungall, C. Torniai, G. V. Gkoutos, S. Lewis, and M. A. Haendel. Uberon, an Integrative Multi-species Anatomy Ontology. *Genome Biology*, 13(1):R5, 2012.

9. S. J. Nelson, W. D. Johnston, and B. L. Humphreys. Relationships in medical subject headings (mesh). In *Relationships in the organization of knowledge*, pages 171–184. Springer, 2001.

10. C. Pesquita, D. Faria, C. Stroe, E. Santos, I. F. Cruz, and F. M. Couto. What's in a "nym"? Synonyms in Biomedical Ontology Matching. In *International Semantic Web Conference (ISWC)*, pages 526–541, 2013.

11. E. Santos, D. Faria, C. Pesquita, and F. M. Couto. Ontology alignment repair through modularization and confidence-based heuristics. arXiv:1307.5322, 2013.

12. L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40(D1):D940–D946, 2012.