On Partitioning for Ontology Alignment*

Sunny Pereira[1], Valerie Cross[1], Ernesto Jiménez-Ruiz[2]

[1] Miami University, Oxford, OH 45056, United States
[2] University of Oslo, Norway

1 Introduction

Ontology Alignment (OA) is the process of determining the mappings between two ontologies. A number of systems currently exists and many of them are participating in the annual Ontology Alignment Evaluation Initiative (OAEI).[3]

Ontology alignment for two very large ontologies becomes time consuming and memory intensive. For example, the *largebio* track in the OAEI campaign still poses serious challenges to participants and only 4 out of 11 systems managed to complete the largest *largebio* task. A general approach to address these challenges is to partition each ontology into cohesive blocks. The matching task is then divided into smaller tasks involving only relevant pair of blocks (i.e., partitions). Ontology partitioning brings new challenges: how best to partition each ontology into blocks and whether the partitioning process on each ontology should be independent of each other. Three main strategies exist: *(i)* totally independent partitioning of both ontologies using various clustering algorithms, *(ii)* independent partitioning of the better structured ontology and then use its partitioning to direct the partitioning of the other, and *(iii)* dependent partitioning between the two using a quick and efficient initial mapping of the two and then this mapping directs their partitioning.

A preliminary study of these three partitioning strategies and their effects on ontology alignment is presented. The objective of this preliminary work is to determine the suitability of these strategies to improve the performance of OA systems when dealing with large ontologies, especially those unable to cope with the largest tasks.

2 Partitioning Algorithms

Partitioning strategies in [3], [4], and [5] all follow a similar method but differ in whether ontology partitioning is done dependent or independent of the alignment task and when the dependence is incorporated. The simplest approach, Partition Block Matching (PBM) [4], first partitions the source and target ontologies separately into blocks. Then I-SUB, an edit-distance based string comparison method, is used on the concepts' labels to determine similarities between the source and target concepts. If the concept labels' string similarity meets a predefined user-settable value in [0,1], then the two concepts become an anchor pair (a_T, a_S).

* This work was partially funded by the BIGMED project (IKT 259055) and the SIRIUS Centre for Scalable Data Access (Research Council of Norway, project no.: 237889)

[3] http://oaei.ontologymatching.org/

Once the anchor pairs are found, a block similarity between each pair of blocks, one from the source and one from the target, is determined using Dice's coefficient calculated as the ratio of the intersection of the anchor pairs between the two blocks b_s and b_t over the sum of the total number of anchors in b_s and the total number of anchors in b_t. A user-settable similarity threshold η in [0,1] must be met between two blocks before marking them as a matched block pair. A block may be paired with more than one block. After block matching, then the alignment between concepts in the matched blocks can begin. Alignment only occurs between the concepts in each matched block pair, not between the whole source and target ontologies.

For dependent partitioning with PAP (partition, anchor, partition) and APP (anchor, partition, partition) [3] anchor pairs are used to direct partitioning of one (PAP) or both (APP) ontologies. If one ontology is more structured than the other, it is first independently partitioned. Then the anchor pairs are determined and used to partition the other ontology (PAP). If not, anchor pairs are first found and used to dependently partition the two ontologies (APP).

For PAP, the first two steps are identical to that of PBM: *(i)* independently partition the more structured ontology O_T, and *(ii)* find anchor pairs between O_S and O_T.The less structured ontology O_S is then partitioned using the blocks b_{T_i} built for O_T in step *(i)*, and the anchor pairs (a_T, a_S) identified in step *(ii)*. Centers CB_{S_i} for a prospective block b_{S_i} in O_S are determined from the anchor pairs existing for b_{T_i}. For each a_T, its corresponding a_S becomes a center CB_{S_i} for a prospective matching block b_{S_i}. A future block b_{S_i} may have multiple centers since multiple anchor pairs may be associated with block b_{T_i}. The centers CB_{S_i} are used to initialize the PBM algorithm for partitioning instead of its simply using each concept in O_S as an individual block. These centers are given the highest cohesiveness value to begin growing the blocks from these centers. A final block b_{S_i} built from a center is matched with the corresponding block b_{T_i}. Not handled by PAP are blocks in O_T and in O_S that have no anchors in them. These blocks are simply ignored and not considered in the mathcing.

The APP method first finds anchors between O_S and O_T. It uses them to partition O_T by favoring the fusion of blocks sharing anchors with O_S. It then partitions O_S by favoring the fusion of blocks sharing anchors with the blocks in the partitioned O_T. The blocks of O_T are generated using PBM but with a modified measure that incorporates not only the strength of the link between blocks b_{T_i} and b_{T_j} within O_T but also the strength of the link of B_{T_j} to O_S as measured by the number of anchors in B_{T_j} relative to the total number of anchors between O_T and O_S. The blocks of O_S are generated by PBM but with another modified measure that uses both the strength of the link between the blocks b_{S_i} and b_{S_j} within O_S and the strength of the link of b_{S_j} to b_{T_k} which is the block in O_T having the highest number of anchors with block b_{S_i}. Blocks of O_S and O_T sharing the highest number of anchors become a matched block pair. One block of O_S can be matched with only one block of O_T. Then alignment between the concepts in each matched block pair is performed.

Table 1. Experiments in *largebio* task 1 suing Wu-Palmer. Matching with LogMap.

Method	FMA Blocks		NCI Blocks		Matching Tasks	Coverage	Precision	Recall	Time (s)	
	#	Isolated	#	Isolated					Partitioning	Matching
PBM	55	15	141	60	87	0.821	0.845	0.743	40.248	85.162
PAP	60	13	141	60	58	0.451	0.870	0.410	39.827	58.517
APP	50	15	143	60	48	0.518	0.870	0.472	41.644	53.157

Table 2. Experiments in *largebio* task 1 using Lin. Matching with LogMap.

Method	FMA Blocks		NCI Blocks		Matching Tasks	Coverage	Precision	Recall	Time (s)	
	#	Isolated	#	Isolated					Partitioning	Matching
PBM	46	6	180	53	83	0.801	0.833	0.728	52.454	81.689
PAP	37	5	180	53	37	0.348	0.861	0.321	56.508	39.423
APP	46	6	180	53	46	0.483	0.862	0.439	56.704	49.938

3 Experimental Methods

The PBM, PAP and APP partitioning methods have been implemented as independent methods from the alignment system. In the preliminary experiments included in this paper we report results for the systems LogMap [6] and FCA-Map [9]. In [3], [4], and [5] a path-based semantic [8] similarity measure is used to determine link strength between concepts within an ontology when creating blocks. In these experiments, the path-based Wu-Palmer [8] as well as information content based Lin [7] semantic similarity measures are considered. The ontology structure is used in determining the information content (IC) for a concept. The link strengths are calculated between concepts that only differ by one in their depth within the ontology. The authors of the PBM method use ISUB to find the anchors between concepts. In our experiments, anchors are found using an exact label match between two concepts in the two different ontologies. Each identified block pair represents a matching (sub)task, however, since blocks are only characterized by a set of concepts, they are first converted to (locality-based) ontology modules [2] and then given to the ontology alignment system as input.

The initial experiments were performed on task 1 of the OAEI *largebio* track,[4] involving small fragments of FMA and NCI, using all three methods. The results using Wu-Palmer are shown below in Table 1 and those for Lin in Table 2. The parameters used are an η of 0.05 for PBM, an α of 0.75 for APP. A maximum block size of 500 and a depth difference of one for semantic similarity calculation is used for all three methods. Blocks with only one concept are considered isolated blocks. *Coverage* represents how many of the entities occurring in the OAEI reference alignments are present in the identified block pairs. The precision and recall are calculated over the combined alignment results for all the matching tasks (i.e., pair of modules extracted from the block pairs). FMA blocks (resp. NCI blocks) represents the number of total blocks produced after partitioning of the FMA ontology (resp. NCI ontology).

The results from task 1 suggest that the PBM method provides much higher recall values than the other two methods. The Wu-Palmer measure performed slightly better than Lin. The next experiments examined how the PBM with the Wu-Palmer performed on the OAEI *largebio* tasks that use the whole ontologies, that is, task 2, task 4 and task 6. The maximum block size is 3000. Table 3 presents these results.

[4] http://www.cs.ox.ac.uk/isg/projects/SEALS/oaei/

Table 3. Experiments with *largebio* whole ontologies using PBM with Wu-Palmer.

Task	System	Source Blocks		Target Blocks		Matching Tasks	Coverage	Precision	Recall	Time (s)	
		#	Isolated	#	Isolated					Partitioning	Matching
FMA-NCI	LogMap	151	2	256	91	69	0.763	0.468	0.675	649	76.7
	FCA-Map							0.506	0.698		$\approx$ 8 hrs
FMA-SNOMED	LogMap	388	9	3352	3273	154	0.594	0.571	0.423	4,807	385
SNOWMED-NCI	LogMap	3357	3160	693	427	443	0.666	0.725	0.491	6,623	937

4 Discussion and future work

In this paper we have presented a preliminary evaluation of state of the art partitioning algorithms for ontology alignment. The obtained results are not good as expected since, after the partitioning and identification of the (sub)matching tasks, the coverage of the entities in the reference alignments is rather low. For example, in the FMA-SNOMED case only 59% of the entities appearing in the reference alignment are covered by the modules in the identified matching tasks. In this case 41% of the entities were lost in either isolated blocks or blocks for which a suitable pair could not be found.

As expected, given the coverage of entities in the reference alignment, the results obtained by LogMap are very low as compared to the results reported for LogMap in last OAEI campaign [1]. In addition the partitioning step represents a considerable overhead with respect LogMap's computation times. Nevertheless, FCA-Map was successfully run in task 2 of the *largebio* track using partitioning,[5] while the system could not cope with the task when given the whole FMA and NCI ontologies [1].

In the close future we aim at investigating new algorithms to provide a suitable partitioning for ontology alignment where the loss of coverage in the identified (sub)matching tasks, in terms of entities of the reference alignments, is minimized. We also intend to perform an extensive evaluation of the novel partitioning algorithms with all OAEI participating systems, especially those failing to cope with the largest tasks.

References

1. Achichi, M., et al.: Results of the Ontology Alignment Evaluation Initiative 2016. In: Proceedings of the 11th International Workshop on Ontology Matching. pp. 73–129 (2016)
2. Grau, B.C., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: Theory and practice. J. Artif. Intell. Res. 31, 273–318 (2008)
3. Hamdi, F., et al.: Alignment-based partitioning of large-scale ontologies. In: Advances in knowledge discovery and management (2010)
4. Hu, W., Qu, Y.: Block matching for ontologies. In: Int'l Sem. Web Conf. (2006)
5. Hu, W., et al.: Matching large ontologies: A divide-and-conquer approach. DKE (2008)
6. Jiménez-Ruiz, E., Grau, B.C.: LogMap: Logic-based and scalable ontology matching. In: Int'l Sem. Web Conf. (2011)
7. Lin, D., et al.: An information-theoretic definition of similarity. In: ICML (1998)
8. Wu, Z., Palmer, M.: Verbs semantics and lexical selection. In: 32nd annual meeting on Association for Computational Linguistics (1994)
9. Zhao, M., Zhang, S.: FCA-Map results for OAEI 2016. In: Proceedings of the 11th International Workshop on Ontology Matching (2016)

[5] Not tested in tasks 4 and 6 due to limited experimental time