Effective method for large scale ontology matching

Gayo Diallo
Univ. Bordeaux, ISPED- LESIM
146 rue Léo Saignat
F-33000 Bordeaux
Gayo.Diallo@isped.u-bordeaux2.fr

Mouhamadou Ba
Univ. Bordeaux, ISPED- LESIM
146 rue Léo Saignat
F-33000 Bordeaux
Mouhamadou.Ba@isped.u-bordeaux2.fr

ABSTRACT
Nowadays, we are facing a proliferation of heterogeneous biomedical data sources accessible through various knowledge-based applications. These data are annotated by more and more large and disseminated knowledge organization systems ranging from simple terminologies and structured vocabularies to very formal ontologies. In order to solve the interoperability issue which arises due to the heterogeneity of these ontologies, an alignment task is usually performed. However, while a significant effort has been undertaken to provide tools that automatically align ontologies containing hundreds of entities, a little attention has been paid to the matching of large size ontologies as it uses to be the case in the life sciences domain. We present in this paper ServOMap, a fast and efficient high precision system able to perform matching ontologies containing hundreds of thousands of entities. The system participated in the 2012 edition of the Ontology Alignment Evaluation Initiative campaign and achieved very good performance, among the top three systems for the Large Biomedical Ontologies Track.

Categories and Subject Descriptors
I.2.4 [Artificial Intelligence] Knowledge Representation Formalisms and Methods– representation languages; H.3.1 [Information Storage And Retrieval] Content Analysis and Indexing - Indexing methods, Thesauruses; J.3 [Life And Medical Sciences]: Medical information systems.

General Terms
Algorithms, Performance, Design.

Keywords
Life Sciences Ontology Matching, Ontology Repository, Semantic Interoperability

1. INTRODUCTION
With the wide adoption of Semantic Web technologies, the increasing availability of knowledge based applications in the life sciences domain raises the issue of finding possible correspondences between the underlying knowledge organization systems (KOS). Indeed, various terminologies, structured vocabularies and ontologies are used for annotating data and the linked open data initiative is increasing this activity. One of the key roles played by these KOS is to provide a support for data exchange based not only on a common syntax but on also on a shared semantic. This particular issue makes them a central component within the Semantic Web and the emerging e-science and e-health infrastructure.

These KOS which are independently developed at the discretion of the various projects are heterogeneous in nature. Moreover, they are becoming more complex, large and multilingual. For instance, the Systematized Nomenclature of Medicine- -Clinical Terms (SNOMED-CT), a multiaxial, hierarchical classification system that is used by physicians and other health care providers for encoding clinical health information, contains more than 300,000 concepts which are regularly evolving. Each concept designated sometimes by several synonymous terms. Another example is the International Classification of Diseases (ICD), the World Health Organization (WHO) standard diagnostic tool for epidemiology, health management and clinical purposes which is used to monitor the incidence and prevalence of diseases and other health problems. The current ICD-10 version contains more than 12,000 concepts designated with terms in 43 different languages including English, Spanish and French.

In many cases, there is a need for establishing mappings between these different KOS in order to make interoperable systems that use them. For instance, the EU-ADR project (1) developed a computerized system that exploits data from eight European healthcare databases and electronic health records for the early detection of adverse drug reactions (ADR). As these databases use different medical terminologies (ICD9, ICD10, Read Code, ICPC) to encode their data, some mappings are needed to translate query posed to the global system into queries understandable by the different data sources. Performing manual mappings between all the mentioned resources is not feasible in a reasonable time. Generally speaking, the data integration domain and the semantic browsing of information domains (2) are areas where matching ontologies is usually performed.

There is, therefore, a crucial need for tools which are able to perform fast and automated correspondences computation between entities of different KOS and which can scale to large ontologies and mapping sets. There is also a need of tools which provide support for multi-ontologies based applications.

Regarding the first issue, a significant effort has been conducted in the ontology alignment/matching domain (3) and the Ontology Alignment Evaluation Initiative campaign has played an important role (4). In this context, it has been noticed during the 2011.5 edition of this campaign that few systems, including GOMMA (5) and LogMap (6), was able to match the whole Foundational Model of Anatomy (FMA) and the National Cancer Institute (NCI) Thesaurus with a good F-measure in a reasonable time.

Regarding the second issue, several initiatives have been conducted in order to provide systems for facilitating accessing multiple and various knowledge artifacts within the semantic web infrastructure (e.g. Swoogle (7), Watson (8), Ontology Lookup Service (OLS) (9) and the BioPortal initiative (10)). However, they follow a centralized approach. Embedding them in an application is not easy as they are not designed with such a purpose.

The work described in this paper falls within the above mentioned research area and presents the ServOMap approach, a large scale ontology matching system which is able to deal with large ontologies associated with multilingual terminologies. ServOMap deals with ontologies described in the RDF(S)[1] and OWL[2] W3C standard languages. It relies on the ServO Ontology Repository (OR) system (11) (12) which is able of managing multiple KOS and provides indexing and retrieving features. Thanks to the use of the ServO OR, ServOMap follows Information Retrieval (IR) based techniques for computing similarity between entities. Contrary to most of the existing large scale matching systems, it is knowledge background free ontology matching system.

From now on, an ontology repository is an index that could be maintained in the memory or in the system files and which store a "representation" of several KOS which are later used for performing some meta-operations including searching similarity between entities. The notion of ontology repository described here differs from the notion represented by system such as OWLIM (13) and more generally Ontology-Based Databases systems (14) and RDF repositories such as Sesame (15). It is more related to the work described in (16).

The rest of the paper is structured as follows. In section 2 we briefly outline the ServO OR on which relies ServOMap and we present its main features. In section 3 we detail the ServOMap ontology matching approach and discribe the different steps for similarity computing. We present in section 4 the evaluation performed on the Large BioMedical dataset provided by the 2012 edition of the OAEI campaign. We conclude in section 5 and give some perspectives as future work.

2. Background on the ServO Ontology Repository

ServO is a system which provides decentralized ontology repository for managing heterogeneous knowledge resources (11). Its design principle is guided by the analogy that could be made between semantic resources retrieval available within an ontology and traditional information retrieval (IR) techniques over a corpus of documents. ServO provides an OR and the functionalities that can be embedded within a knowledge-based application for accessing the managed ontologies.

It provides functionalities to meet the following set of requirements:

- allowing building and maintaining decentralized repositories and make them communicating
- providing the ability to dynamically index a set of ontologies in a single repository that can be later updated as needed
- be able to overcome the difference in the languages used for describing ontologies

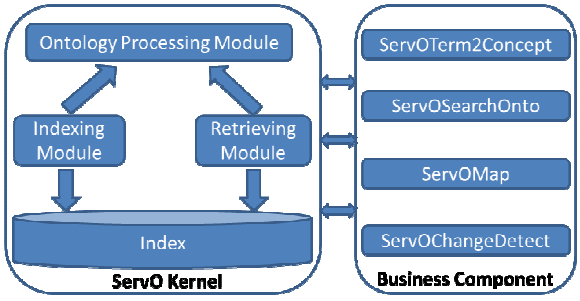

Figure 1: The Servo Kernel and Business Components (meta operations)

Thus, the approach adopted is based on the adaptation of IR tested and validated methods. And the following choices have been made (figure 1). First, a common meta-model is defined for representing any ontology regardless its language or format. This meta-model is instantiated by processing the input ontology with the JENA framework (17). Then, an Ontology Processing and loading module is designed and implemented. Finally, an Ontology Indexing Module (OIM) and an Ontology Retrieving Module (ORM) are designed.

The OIM and the ORM use the high-performance scalable information retrieval library Apache Lucene[3]. These components are detailed in (11).

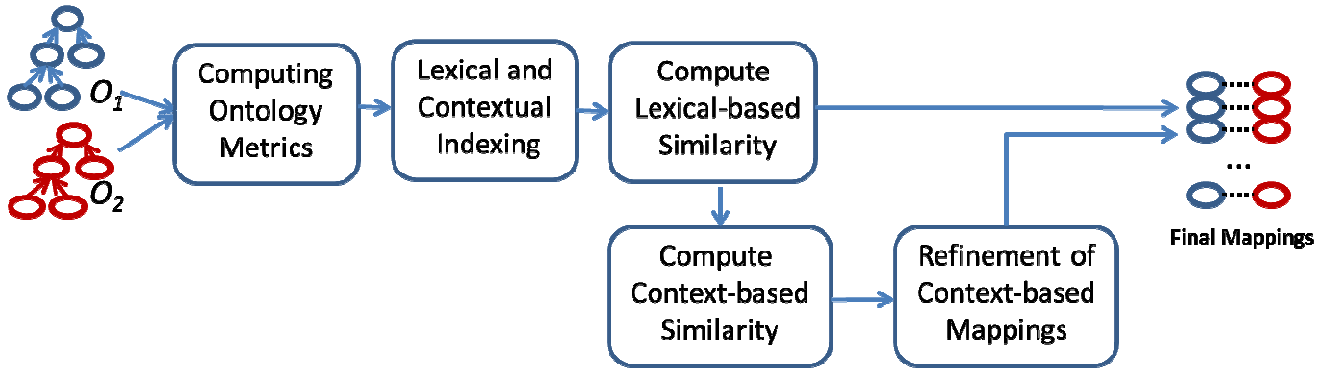

Figure 2: Overview of the ServOMap approach

The model for the OR defines the two main functionalities of the repository: indexing and retrieving resources according to some criteria. An indexing and retrieval model specifies how

[1] http://www.w3.org/TR/rdf-schema/

[2] http://www.w3.org/TR/owl-features/

[3] http://lucene.apache.org

documents and queries must be represented. Also it details the retrieval function to be used. Moreover it determines the notion of relevance. The relevance can be binary (the case of the Boolean model) or continuous (a ranked list of results).

ServO allows querying the repository by combining Boolean terms (a.k.a the labels of the entities) and both datatype and object properties. This requirement allows comparing in a structured basis several concepts from different ontologies. Following the functionality offered by the Lucene API, we adopted an approach which combines both the Boolean and the Vectorial space models (VSM) of IR to compute the relevance between the queries and the entities of the ontologies within the repository.

In the VSM, each document or query is represented by a vector in a space where each dimension is associated to an indexing term. The similarity between the query q and the concept c is computed as (11):

$$score(q,c) = coord(q,c) \cdot queryNorm(q) \cdot \sum(tf(t \text{ in } c) \cdot idf(t)^2 \cdot t.getBoost() \cdot norm(t,c))$$

Where:

- tf(t in c) correlates to the term's frequency, defined as the number of times term t appears in the currently scored concept c. tf(t in c) = √frequency
- icf(t) stands for Inverse Concept Frequency. This value correlates to the inverse of ConceptFreq (the number of concepts in which the term t appears).
- coord(q,c) is a score factor based on how many of the query terms are found in the specified concept.
- queryNorm(q) is a normalizing factor used to make scores between queries comparable. It attempts to make scores from different queries (or even different indexes) comparable.
- t.getBoost() is a search time boost of term t in the query q as specified in the query text.
- norm(t,c) encapsulates a few (indexing time) boost and length factors such as Concept boost and Field boost.

Finally, the different functionalities offered by the ServO OR are:

- Mapping users query terms to concepts from previously indexed ontologies (Term2Concept)
- Ontology matching and semantic similarity computing between entities for different ontologies (ServOMap)
- Ontology searching in order to provide a KOS or a set of KOS suitable for a particular task (ServOSearch)
- Change detection between different versions of the same KOS (ServOChangeDetect).

In the following section, we detail the ontology matching process ServOMap which is based on the use of the ServO OR.

3. Large scale ontology matching with ServOMap

In this section, we detail the overall process that ServOMap follows for computing similarity between entities of two given ontologies and more generally two given knowledge organization systems. The approach is depicted in Figure 2. There are 5 steps that are described below.

3.1 Computing Ontology Metrics

The first step after parsing and loading input ontologies is to compute a set of metrics that are later used as parameters for the systems. These metrics include for any input ontology: the average number of sub-concepts for a concept, the different languages used to denote entities labels or annotations, the most frequent terms within the ontology, the longest set of synonyms labels used to describe a concepts, etc. Some metrics are necessary for optimizing the use of the Lucene backend.

3.2 Lexical and Contextual Indexing

As we have already pointed out, ServOMap relies on IR techniques for ontologies matching. Therefore, an ontology is seen as a corpus of document to process. Each entity (concepts, properties including both object properties and data type properties) is a document to process.

To do so, ServOMap constructs an inverted index (an ontology repository) from the input ontologies. Thus, for each ontology, ServOMap uses the Ontology Processing Module of ServO to retrieve all entities (concepts and properties). Then, according to the parameters computed during the previous step (Computing Ontology Metrics) a dynamic generation of entity description is performed. This process is dynamic as each entity is described according to the features it holds. Thus, some concepts may have synonyms in several languages or may have comments where other may only have English terms. Though, some concepts may have declared properties (either object properties or datatype properties), etc. During the dynamic description process, the retrieved labels from a concept are passed to a set of filters: stop words removal, normalization (upper case to lower case), punctuations removal, completion of labels by the permutations of their terms and so on. It is also possible to indicate whether ServOMap uses label stemming or not. Moreover, the words of a term can be concatenated as in the Table 1.

TABLE I. EXAMPLE OF AVAILABLE FIELDS WITHIN THE INDEX AND THEIR TERM COUNTS FOR THE FOUNDATIONAL MODEL OF ANATOMY ONTOLOGY

Field Name	Term Counts	Example
dDomain	15	spatialassocirelat
dRange	5	string
directLabelCEn	152,088	accessorilobarvein veinaccessorilobar veinlobaraccessori
directNameC	78,884	accessorilobarvein
directNameP	52	percentag
uri	79,042	http://bioontology.org/#Accessory_lobar_vein

Table 1 gives an example of available fields and their term counts within the index for the Foundational Model of Anatomy ontology (FMA). Term counts are provided by the Lucene backend. FMA contains 79,042 entities, among them 78,884 are concepts. As we can see, the value of the *dDomain* field (the domain of a property) is spatialassocirelat which is the term "spatial association relation". And the concept with id #Accessory_lobar_vein has as *directLabelCEn* the set {*accessorilobarvein veinaccessorilobar veinlobaraccessori*} for "Accessory lobar vein" and its permutations. All spaces are removed within labels.

In ServoMap we make the assumption that two concepts similar have likely their surrounding concepts similar. Thus, the description of a concept is completed by contextual descriptions. The first one is the SubConcept strategy where a concept is completed by the information about all its sub-concepts. The second strategy is the SupConcept strategy where each concept is completed by the description of its super-concepts. The third one is the SibConcept strategy. In this case the description of a concept is completed by the description of all its siblings.

A flag is used to indicate whether the two input ontologies have to be indexed or only the smallest one. This flag is exploited latter during the similarity computing phase.

3.3 Compute lexical based similarity

After the indexing phase, ServOMap proceeds to the lexical based similarity computing. This step relies on the Ontology Retrieval Module of the ServO Ontology Repository and use the similarity function described in section 2.

Depending on the flag indicating the indexed ontologies, the Ontology Processing Module is called for retrieving the concepts to use for searching over the built index. Thus, if both input ontologies are indexed, the first one, let's say O_1 is used as search ontology over the index on the second ontology I_2. And, vice versa, the ontology O_2 is used to perform search over the index of the first ontology I_1. If the flag indicates that only one ontology is indexed, then ServOMap performs only a one way search.

As in the lexical and contextual indexing phase, a dynamic generation of entity description if performed for any entity to use in order to search the index. A Boolean query is constructed with all the available fields for the entity (label, comments, properties, etc.). Please note that the same string processing task is performed for all the components of the entity in order to have the same level of description than the indexing phase.

Again, ServOMap relies on the ServO OR. Each Boolean query represented as a vector of terms is searched over the index. A ranked list of entities is retrieved. SeroMap keeps as a possible mapping the couple constituted of the entity to search and the entity having the highest similarity (vectorial similarity) with the entity to search. It can happen that several entities have the same similarity with the entity to search. In this case, in order to keep the most relevant, the local names of the entities are compared using the Levenshtein Distance.

At the end of this process, a first set of mappings between the two ontologies is made available.

3.4 Compute context-based similarity

Usually the mappings computed previously are considered high precision based mapping. Indeed, as it is almost a strict equality that is used between entities to compare, and only the direct description is used, the mapping is likely to be correct. However, this high-level accuracy is relativized by the relatively low recall. And, as the objective is to return as many mappings as possible, there is a need to complete the set of mappings obtained previously.

To do so, a contextual based similarity is performed. The idea is based on the assumption that when two entities are similar, there is a big chance that the concepts that surround it are also similar. Here, by surrounding concepts we mean super-concepts, sub-concepts and siblings concepts. Thus, in the context based similarity, the description of a concept is based on the strategies outlined previously (i.e. SubConcept, SupConcept, SibConcept). This contextual strategy is applied only on concepts and not on properties. And, it is restricted to only the concepts that have not been yet mapped to any other concepts. This is again based on the assumption that if two concepts are mapped by the previous strategy, it is likely to be correct.

The same process as previously is followed for dynamically generating the description of the concepts. The resulting query is sent to the index for retrieving the possible mappings. The same process is repeated for *SubConcept*, *SupConcept*, *SibConcept*.

After the complete process, we have three sets of mappings according to the three strategies. These three sets are then combined and duplicates mappings are removed.

As our approach is mainly lexical based, we realized during our experiments that this strategy generates a lot of noise. We then defined a refinement strategy to select the best mappings among the set obtained during the context based mapping. This strategy is briefly described in the following section.

3.5 Refinement strategy for context-based mappings

During the context mappings refinement we try to keep only the couples obtained and that do not contradict the mappings that are already found with the lexical based mappings. Again, here, this is based on the assumption that the lexical-based similarity is highly accurate. In order to filter out the results provided by the *SubConcept, SibConcept, SupConcept* strategies we use the refinement algorithm described above and illustrated in figure 3. In this figure, *ContextM* is the set of mappings provided by the context-based

```
Algo Refinement_SubSupSib
input: vector ContextM, LexicalM
output: vector CleanContextM
Begin
For each couple (C1, C2) in ContextM
    If C1 OR C2 exists in LexicalM Then
        1. If C1 is LexMappedWith Sup(C2) or Sub(C2) Or
           C2 is LexMappedWith Sup(C1) or Sub(C1)) Then
                removeCouple(C1,C2)
        2. If C1 is LexMappedWith Sib (C2) Then
                removeCouple(C1,C2)
        3. If C2 is LexMappedWith Sib (C1) Then
                removeCouple(C1,C2)
        4. If Sub(C1) isMappedWith (Sib(C2) OR Sup(C2)
                Then removeCouple(C1,C2)
        5. If Sup(C1) isMappedWith (Sib(C2) OR Sub(C2)
                Then removeCouple(C1,C2)
        Do 4.) and 5.) for C2
    EndIf
EndFor
        return CleanContextM ;
End
```

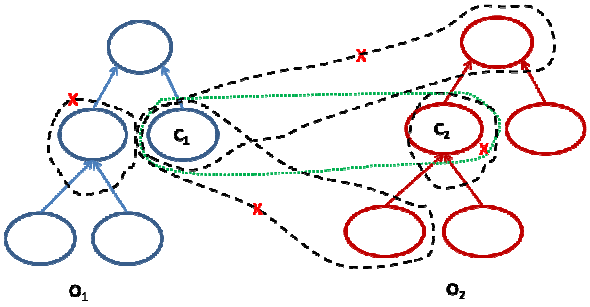

Figure 3: Refinement strategy. If C1, C2 is obtained from the lexical mapping, all the contextual-based mappings which contradict C1, C2 are removed

strategy; *LexicalM* is the set of mappings computed by the lexical based strategy. The idea is to avoid keeping a couple obtained from the context based similarity where one of the entries is already mapped during the lexical process by another concept. This strategy takes into account the worst case and allows removing several unwanted mappings and increase the recall at the same time. However, it generates noise, and the precision obtained with lexical-based mappings is then reduced.

3.6 Processing Disjoints Concepts

Some knowledge organization systems are described in formal languages allowing expression complex axioms and constraints. In particular, declared disjoints concepts can be

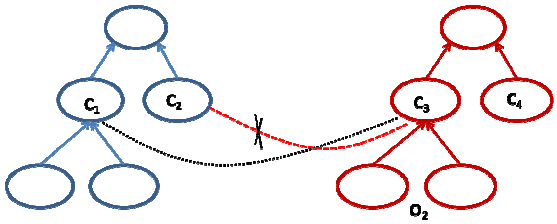

Figure 4: Strategy for processing disjoints concepts

found in certain KOS. As our approach is mainly based the lexical description of the features of entities, it is possible to find two concepts lexically similar while they are semantically declared as disjoint. In order to avoid such a situation, we have taken into account these cases during both indexing and retrieving phases.

Let's assume that C_1 and C_2 are two disjoints OWL concepts belonging to an ontology O_1 and C_3 and C_4 two other disjoints concepts belonging to the ontology O_2 (figure 4). In order to compute the similarity between C_1 and C_3, we proceed as follows:

- If it is O_2 which is indexed, then C_3 must have a field *Disjoint_Concept* which contains all the generated description terms of C_4. ServOMap proceeds inversely if O_1 is indexed
- During the similarity computing phase, when the score between C_1 and C_3 is computed, the query is built taking into account the fact no terms from the field *Disjoint_Concept* of C_1 (i.e. C_2) appears in the generated description of C3. Similarly, no terms from the *Disjoint_Concept* field of C_3 (i.e. C_4) appears in the generated description of C_1. Thus, we ensure a set of coherent mappings regarding disjointnes.

In the following section we present the evaluation of ServOMap that has been performed on a set of various dataset.

4. Evaluation

In this section, we report the performance achieved by our system on the large biomedical track of the OAEI 2012 campaign. To do so, we will describe first OAEI and the dataset that has been used in our evaluation.

4.1 The Ontology Alignment Evaluation Initiative

The Ontology Alignment Evaluation Initiative known as the OAEI campaign is an international campaign for the systematic evaluation of ontology matching systems. A matching system is defined by OAEI as a software programs capable of finding correspondences (called alignments) between the vocabularies of a given set of input ontologies (3). The campaign started in 2004 and is mainly motivated by the need to establish a consensus for the evaluation of the ever increasing number of methods available for schema matching or ontology integration. It is usually associated with Ontology Matching (OM) Workshop of the International Semantic Web Conference (ISWC).

For the 2012 edition[4] of the campaign there were 23 participating systems for six entity matching problems and three others for the instance matching problem. This edition was aiming at automated evaluation to a large extent with new test sets that have been made available. This is the case with the Large Biomedical ontologies track referred to as LargeBio described in the next section.

The SEALS platform (18) is used for the automated evaluation of all the systems. The SEALS project is dedicated to the evaluation of semantic web technologies. It created a platform[5] for easing this evaluation, organizing evaluation campaigns, and building the community of tool providers and tool users around this evaluation activity. The different participant systems are wrapped according to the SEALS specification before to be uploaded to the platform. The overall process for the OAEI 2012 campaign using this platform is described in the campaign web site[6].

4.2 The OAEI 2012 LargeBio dataset

The LargeBio track is one of the most challenging tasks in term of scalability and complexity. The ontologies in this dataset are semantically rich and contain tens of thousands of classes. Indeed, the track consists of finding alignments between the Foundational Model of Anatomy (FMA) which contains 78,989 concepts, the SNOMED-CT which contains 306,591 concepts, and the National Cancer Institute Thesaurus (NCI) which contains 66,724 concepts.

The FMA is a domain ontology that represents a coherent body of explicit declarative knowledge about human anatomy. It is integrated in the distributed framework of the Anatomy Information System developed and maintained by the Structural Informatics Group at the University of Washington It is concerned with the representation of classes or types and relationships necessary for the symbolic representation of the

[4] http://oaei.ontologymatching.org/2012/

[5] http://www.seals-project.eu/

[6] http://oaei.ontologymatching.org/2012/seals-eval.html

phenotypic structure of the human body in a form that is understandable to humans and is also navigable, parseable and interpretable by machine-based systems.

SNOMED CT is a clinical healthcare terminology which provides a core general terminology for the electronic health record (EHR) and contains currently more than 311,000 active concepts with unique meanings and formal logic-based definitions organized into hierarchies. It is owned, maintained and distributed by the International Health Terminology Standard Development Organization (IHTSDO).

The NCI Thesaurus covers vocabulary for clinical care, translational and basic research, and public information and administrative activities. It provides reference terminology for many National Cancer Institute of the US National Institutes of Health and other systems.

The LargeBio track consisted of three matching problems: FMA-NCI matching problem, FMA-SNOMED matching problem and SNOMED-NCI matching problem. Each matching problem is divided in three tasks involving different fragments of the considered ontologies, i.e. a small fragment of the ontologies, a big fragment and the whole ontologies. This leads to 9 sub-tasks. The 2009AA version of the Unified Medical Language System (UMLS) Metathesaurus is used as the basis for the track reference alignments (19).

4.3 The configurations used for ServOMap

As ServOMap is highly flexible, it participated in the campaign with two configurations. They differ by the parameters that are used to tune the matching process. These parameters are depicted on Table 2. The first version of the system that we refer to as ServOMap-lt uses the same processing technique for the terms of the entities being matched regardless their language (English, French, etc.).

TABLE II. TABLE 1: PARAMETERS USED TO TUNE THE TWO VERSIONS OF THE SYSTEM

	ServOMap-lt	ServOMAP
Terms processing	The same for all languages	According to the language of the labels
Entities taken into account	Only Concepts	All Entities
Ontologies indexed	One	Both
Searching strategy	One way	Two ways
Stemming	Yes	No
Arity of the mappings	1:n	1:1

In addition, only concepts are taken into account contrary to the second version, which we refer to as ServOMap. Also, only one of the input ontology is indexed with ServOMap-lt, the second one being used for searching over the index. Finally, ServOMap-lt uses stemming techniques for the labels and it performs 1:n mappings while ServOMap takes into account only 1:1 mappings and does not use stemming. The two versions are freely available for download online[7].

4.4 Results

The evaluation is performed in a server with 16 CPUs and allocating 15 Gb RAM. 15 out of 23 participating systems/configurations have been able to cope with at least one of the tasks of the LargeBio track matching problems.

TABLE III. SERVOMAP-LT PERFORMANCE ON THE LARGEBIO DATASET

Task	Precision	Recall	F1-measure	Time (s)
FMA-NCI	0.931	0.8	0.86	366
FMA-SNOMED	0.956	0.60	0.802	790
SNOMED-NCI	0.875	0.593	0.706	1,248
AVERAGE	0.890	0.699	0.780	2,405

The performance of the two versions of the ServOMap system is depicted on Table 3 and 4. We have averaged the results obtained on the entire sub-tasks (small, big, and whole). We refer the reader to the OAEI 2012 LargeBio web page for the complete results of the evaluation[8]. Thus, each matching problem (FMA-NCI, FMA-SNOMED, SNOMED-NCI) is presented in one row. The last entry gives the average of the entire LargeBio track. The last column gives the total computation times.

TABLE IV. SERVOMAP PERFORMANCE ON THE LARGEBIO DATASET

Task	Precision	Recall	F1-measure	Time (s)
FMA-NCI	0.945	0.747	0.834	327
FMA-SNOMED	0.953	0.656	0.777	893
SNOMED-NCI	0.901	0.554	0.687	1,089
AVERAGE	0.903	0.657	0.758	2,310

The best precision is obtained for the FMA-SNOMED matching problem with 95.6% and 95.3% for ServOMap-lt and

[7] http://code.google.com/p/servo/

[8] http://www.cs.ox.ac.uk/isg/projects/SEALS/oaei/2012/results2012.html

ServOMap respectively. The best recall is obtained for the FMA-NCI matching problem. ServOMap-lt obtained 80% while ServOMap obtained 83.4%. We can notice on average that ServOMap-lt provides the best recall (65.7%) while ServOMap achieves the best precision (90.3%). Clearly, these results show that ServOMap-lt benefited from 1:n mappings by providing more correspondences that can be found in the reference alignment. However, this decreased its precision. Another explanation of the lower precision is the use of stemming techniques which lead to grouping to the same index entry different labels having the same stem. In contrast, ServOMap thanks to the 1:1 mapping strategy was able to provide the most precise correspondences, but with a lower recall.

From the computation time point of view, the SNOMED-NCI task was the longest to complete with respectively 1,248 seconds (20.8mn) and 1,089 seconds (18.15mn) for ServOMap and ServOMap-lt. In contrast, the FMA-NCI matching problem was the fastest to complete. ServOMap-lt performed the task in 366 seconds (6.1mn) while ServOMap finished in 327 seconds (5.45mn). These results are in line with the size of the ontologies to match. The SNOMED-NCI is the largest task to process in term of involved entities.

Now let's compare our system to the other participating systems which completed the LargeBio track. According to the official OAEI results, we have presented the summary of the top-8 systems in Table 5. According to these figures, ServOMap-lt provided the best results in terms of F-measure and precision for the FMA-SNOMED task while ServOMap generated the most precise mappings when all the task are averaged, with 90.3%. ServOMap-lt finished overall second in term of F-measure with 78% closely behind the YAM++ system (78.2%) (20). For the computation times, ServOMap finished the entire 9 tasks in 2.310 seconds (38.5 mn) at the second position behind the LogMaplt system (711 seconds) (14) while YAM++ completed them in 18 hours. We mention that GOMMA, YAM++ and LogMap systems use different kinds of background knowledge. LogMap uses normalisations and spelling variants from the UMLS Lexicon while use the general purpose background knowledge provided by WordNet and GOMMA reuses mappings from FMA-UMLS and NCI-UMLS.

Please note that the last column of table 5 (Incoherence) reports the number of unsatisfiabilities when reasoning using the HermiT reasoner with the input ontologies together with the computed mappings. The logic assessment of computed mappings is not a feature implemented yet into ServOMap. LogMap was the system which provides the cleanest mappings.

TABLE V. SUMMARY RESULTS OF THE LARGEBIO TOP 8 SYSTEMSTRACK

System	Total Time (s)	Average			
		Precision	Recall	F-measure	Incoherence
YAM++	67,817	0.876	0.710	**0.782**	45.30%
ServOMapL	2,405	0.890	0.699	0.780	51.46%
LogMap-noe	3,964	0.869	0.695	0.770	**0.004%**
GOMMA_Bk	5,821	0.767	**0.791**	0.768	45.32%
LogMap	3,077	0.869	0.684	0.762	**0.006%**
ServOMap	2,310	**0.903**	0.657	0.758	55.36%
GOMMA	5,341	0.746	0.553	0.625	24.01%
LogMapLt	**711**	0.831	0.515	0.586	33.17%

5. Conclusion and Perspectives

We have presented in this paper the main component of the ServO Ontology Repository and detailed its ServOMap component for large scale ontology matching. We have reported the performance obtained by this component on the LargeBio track during the 2012 edition of the OAEI campaign. The two versions of ServOMap achieved very good results both in term of F-measure and computation times by finishing among the top-3 systems and providing mappings with the best precision. We notice, however, that so far our approach relies heavily on the richness of the description of the input ontologies, which used to be the case in the life sciences domain. The efficiency is reduced for KOS whose mappings must be based more on the structure.

There is a room of improvement of this research work. First, we plan to improve the algorithm used for filtering out the mappings provided by the context-based matching in order to increase recall without reducing the precision. ServOMap does not use any external resource in the similarity computing process. We intend to use the UMLS resource for better discarding wrong mappings for the ontologies presents in this resource. Moreover, the current version does not take into account the mapping of two ontologies described in two different languages. For instance, an ontology with terms in English to compare with an ontology with terms in German. An improvement of the system is then to implement a cross lingual ontology matching. Finally, we plan introducing logic assessment of computed mappings (21) and implementing a user-friendly interface.

6. Acknowledgment

We thank the organizers of the OAEI evaluation campaigns for providing us the test data and Seals infrastructure and the LargeBio track organizers for their valuable feedback.

7. References

1. Avillach P, Mougin F, Joubert M, Thiessard F, Pariente A, Dufour J-C, et al. A semantic approach for the homogeneous identification of events in eight patient databases: a contribution to the European eu-ADR project. Stud Health Technol Inform. 2009;150:190–4.

2. Diallo G, Khelif K, Corby O, Kostkova P, Madle G. Semantic Browsing of a Domain Specific Resources: The Corese-NeLI Framework. Web Intelligence/IAT Workshops. 2008. p. 50–4.

3. Shvaiko P, Euzenat J. Ten Challenges for Ontology Matching. In: Meersman R, Tari Z, editors. On the Move to Meaningful Internet Systems: OTM 2008 [Internet]. Springer Berlin / Heidelberg; 2008. p. 1164–82. Available from: http://dx.doi.org/10.1007/978-3-540-88873-4_18

4. Euzenat J, Meilicke C, Stuckenschmidt H, Shvaiko P, Santos CT dos. Ontology Alignment Evaluation Initiative: Six Years of Experience. J. Data Semantics. 2011;15:158–92.

5. Kirsten T, Gross A, Hartung M, Rahm E. GOMMA: a component-based infrastructure for managing and analyzing life science ontologies and their evolution. Journal of Biomedical Semantics. 2011;2(1):6.

6. Ruiz EJ, Grau BC, Zhou Y, Horrocks I. Large-scale Interactive Ontology Matching: Algorithms and Implementation.

Proceedings of the 20th European Conference on Artificial Intelligence (ECAI). IOS Press; 2012. p. 444–9.

7. Finin T, Peng Y, Scott R, Joel C, Joshi SA, Reddivari P, et al. Swoogle: A search and metadata engine for the semantic web. In Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management. ACM Press; 2004. p. 652–9.

8. d' Aquin M, Motta E, Sabou M, Angeletou S, Gridinoc L, Lopez V, et al. Toward a New Generation of Semantic Web Applications. IEEE Intelligent Systems. 2008;23:20–8.

9. Côté RG, Jones P, Apweiler R, Hermjakob H. The Ontology Lookup Service, a lightweight cross-platform tool for controlled vocabulary queries. BMC Bioinformatics. 2006;7:97.

10. Noy NF, Shah NH, Whetzel PL, Dai B, Dorf M, Griffith N, et al. BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research. 2009 May 29;37(Web Server):W170–W173.

11. Diallo G. Efficient Building of Local Repository of Distributed Ontologies. IEEE; 2011 [cited 2012 Oct 6]. p. 159–66. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6120644

12. Diallo G. Towards decentralized and cooperative repositories of distributed ontologies. Proceedings of SWAT4LS 2011. 2011. p. 8–9.

13. Kiryakov A, Damova M. The Semantic Web: Semantic Repositories. Semantic Web Handbook. Springer Verlag, Heidelberg Germany. 2011.

14. Fankam C, Jean S, Pierra G, Bellatreche L, Ait-Ameur Y. Towards Connecting Database Applications to Ontologies. IEEE Computer Society, Conference Publishing Service; 2009. p. 131–7.

15. Schenk S, Petrak J. Sesame RDF Repository Extensions for Remote Querying. Znalosti2008 [Internet]. 2008. Available from: http://znalosti2008.fiit.stuba.sk/download/articles/znalosti2008-Schenk.pdf

16. Ghoula N, Falquet G. Towards an ontology based large repository for managing heterogeneous knowledge resources. E-LKR'12. 2012.

17. Carroll JJ, Dickinson I, Dollin C, Reynolds D, Seaborne A, Wilkinson K. Jena: implementing the semantic web recommendations. Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters [Internet]. New York, NY, USA: ACM; 2004. p. 74–83. Available from: http://doi.acm.org/10.1145/1013367.1013381

18. Esteban-Gutiérrez M, Garcıa-Castro R, Gómez-Pérez A. Executing Evaluations over Semantic. Technologies using the SEALS Platform. IWEST 2010. 2010.

19. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Research. 2004;32(Database-Issue):267–70.

20. Ngo D, Bellahsene Z. YAM++ : A Multi-strategy Based Approach for Ontology Matching Task. In: ten Teije A, Völker J, Handschuh S, Stuckenschmidt H, d' Aquin M, Nikolov A, et al., editors. EKAW [Internet]. Springer; 2012. p. 421–5. Available from: http://dblp.uni-trier.de/db/conf/ekaw/ekaw2012.html#NgoB12

21. Meilicke C, Stuckenschmidt H, Sváb-Zamazal O. A Reasoning-Based Support Tool for Ontology Mapping Evaluation. ESWC. 2009. p. 878–82.