

Web Services Matchmaking Based on a Partial Ontology Alignment

Aissa FELLAH

EEDIS Laboratory, Djillali Liabes University of Sidi Bel Abbes, Algeria
Laboratoire de Recherche d'Informatique de Sidi Bel Abbes: LabRI-SBA Laboratory
E-mail: ammfellah@yahoo.fr

Mimoun Malki

Ecole Supérieure d'Informatique de Sidi BEL-ABBES (ESI-SBA), Algeria
Laboratoire de Recherche d'Informatique de Sidi Bel Abbes: LabRI-SBA Laboratory
EEDIS Laboratory, University of Djillali Liabès, Sidi BEL-ABBES, Algeria
E-mail: m.malki@esi-sba.dz

Atilla ELÇI

Dept. of Electrical-Electronics Engineering, Aksaray University, Turkey
E-mail: atilla.elci@gmail.com

Abstract—The fast development and the huge number of existing web services have raised the problem of the urgent need for matchmaking mechanisms. However state-of-the-art matchmakers are unsuitable for locating web services that use different ontologies. This aspect is important since it is not realistic to assume that Web services will always be defined by the same ontology, as the Web service requester and provider operate independently, each defines their own ontologies to describe their services. This is an emergent research issue that has not been well addressed. This work is a contribution to achieve semantic interoperability in a multi-ontology environment. This paper describes a Web service multi-ontology matchmaker for SAWSDL services, called SAWSDL-MOM which locates web services that use different ontologies. The matchmaker engine incorporates a novel partial ontology alignment algorithm with syntax, linguistic and original structural matchers. In determining the 1:1 mappings the Hungarian algorithm is used. Finally a matchmaking strategy is utilized in finding the score of each service. Experimental evaluation and comparison provide strong evidence that SAWSDL-MOM can significantly improve results, achieve better interoperability and scalability.

Index Terms—Semantic Web, SAWSDL, Semantic Service Matchmaking, Partial Ontology Alignment, Web Services Interoperability.

I. INTRODUCTION

W3C defines 'Web service discovery' as "the act of locating a machine processable description of a Web service that may have been previously unknown and that meets certain functional criteria. It involves matching a set of criteria with a set of Web service descriptions. The

goal is to find an appropriate Web service." [1] Absence of semantics in description creates inability in exploiting the Web service discovery. Describing Web service with semantics provides the potential for automatic Web service discovery, invocation, composition and interoperation. Semantic Web Services (SWS) extend the idea of the Semantic Web to Web Services (WS). SWS aim to complement the current knowledge-poor syntactic industry standards with semantic metadata in order to facilitate automation of WS related tasks. Many papers have discussed semantic matching, when advertisement and request use the same ontology but an approach based on using a unique ontology is not rational because it requires that every service provider and requestor should adhere to the same ontology, an assumption which is improbable. In open distributed computing environment available mechanisms for semantic service discovery face new challenges such as increasing scale of systems and multiple coexisting ontologies [2][3]. Semantic alignment mechanisms need to be purposefully integrated into a service discovery framework in order to fully exploit its potential. Measuring semantic similarity between SWS may be reduced to the mapping between ontologies. Ontologies are used to provide semantic interoperability, but they themselves may be heterogeneous. One of the most promising and mature approaches achieving that interoperability is ontology matching [4]. It establishes relations between terms of different ontologies by calculating semantic similarity between them. In order to achieve web services matchmaking in a multi-ontology environment we propose a framework (see Fig.1) that supports for input of a query and web service both expressed in SAWSDL each with its own ontology. The matching part is mainly based on a partial ontology alignment module which uses a similarity measure between concepts across ontologies. Partial alignment results support a matchmaking strategy.

This paper is organized as follows. Section 2 presents useful concepts for the rest of the document and provides an overview of related works. This helps clarify ideas and expose our approach. Section 3 presents the main contribution of this article: based explicitly on the partial alignment between ontologies, a matching approach for SAWSDL web services which may be annotated by different ontologies. Section 4 presents experimental evaluation of our ideas. Finally, in Section 5, we conclude and propose work.

II. RELATED WORK

In this section, we begin by presenting a brief background which is important to follow the rest of the document. Next, we describe related work, which we divided in two parts: the first is a classification of matchmaking approaches, where we categorize proposed solutions into two categories, one for solutions that support a single ontology and another that support multi-ontology solutions. In the second part of our overview we describe a brief survey of ontology matching techniques.

A. Background

1. SAWSDL

SAWSDL[5] (Semantic Annotations for WSDL) is the most popular Semantic Annotations for WSDL, becoming a W3C recommendation, enabling service providers to enrich their service descriptions with additional semantic information. SAWSDL introduces three varieties of attributes for the semantic annotation: `modelReference`, `liftingSchemaMapping` and `loweringSchemaMapping`, which are used to annotate existing Web services described in WSDL. A `modelReference` points to concepts with equally intended meaning expressed in an arbitrary semantic representation language. They are allowed to be defined for every WSDL and XML Schema element, though the SAWSDL specification defines their occurrence only in WSDL interfaces, operations, faults as well as XML Schema elements, complex types, simple types and attributes. The purpose of a `modelReference` is mainly to support automated service matchmaking. The SAWSDL specification does not restrict the type of semantic concepts a `modelReference` should point to. The only requirement is that the concepts are identifiable via URI references. In the context of this work, we will assume that annotations are only with OWL ontology concepts. As a second constraint, SAWSDL-MOM checks only the first `modelReference` of an element.

2. Ontology and Alignment

Ontology is a recent knowledge representation technique. It is identified as the base technology for the Semantic Web which is a wide view for the future development of WWW [6] and it is used as the formalized domain knowledge specifications for SWS descriptions and general semantic programming. Ontologies are employed extensively in numerous fields

such as knowledge engineering, artificial intelligence and applications related to knowledge management, information retrieval, linked data and the semantic Web. Ontology can be defined as a 4-tuple: $O = (C, R, H, I)$ where C represents the set of concepts in ontology; $R \subseteq C \times C$ is the set of relations over concepts; $H \subseteq R$ is a subset of R which represents hierarchical relation set between concepts; and I is a set of instances. A concept is composed of: 1- A URI (Unique Resource Identifier), 2- A set of names (comment, name, and label), 3- Internal context (a set of internal properties) where each property consists of: Name, Range, and Domain, 4- External context composed of Fathers (set of concepts), Childs (Set of concepts), and Brothers (Set of concepts). An alignment (or mapping) between two ontologies O and O' can be described as a quadruple [7] (e, e', n, R) where: - e and e' are the entities between which a relation is asserted by the mapping (e.g., formulas, terms, classes, individuals), - n is a degree of trust (confidence) in that mapping, and, - R is the relation associated to a mapping, where R identifies the relation holding between e and e' . A mapping between two models rarely maps all the concepts in one model to all concepts in the other. Instead, mappings typically loose some information and can be partial or incomplete [8]. A total ontology mapping from $O_1 = (S_1, A_1)$ to $O_2 = (S_2, A_2)$ is a morphism $f: S_1 \rightarrow S_2$ of ontological signatures, such that, $A_2 = f(A_1)$, which means that all the interpretations that satisfy O_2 satisfy O_1 . Of course, in reality, it is difficult to attain these total mappings and therefore there is the notion of a partial ontology mapping from $O_1 = (S_1, A_1)$ to $O_2 = (S_2, A_2)$ if there exists a sub ontology $O'_1 = (S'_1, A'_1)$ where S'_1 subset of S_1 and A'_1 is a subset of A_1 such that there is a total mapping from O'_1 to O_2 .

B. Related Work

To the best of our knowledge, there is no other approach that uses the partial ontology alignment technique for Web services discovery. In this work we are concerned with the use of ontology alignment techniques in Web service matchmaking. For this, we believe that a study of alignment techniques and matchmaking in single and multi-ontology is useful in our context. For a survey of semantic service matchmakers in general, we refer the interested reader to [23].

1. Matching Web Services Using Single Ontology

LARKS "Language for Advertisement and Request for Knowledge Sharing" is the first contribution conducted by Sycara et al. that regards matchmaking in the context of SWS [9]; it has heavily influenced the SWS matchmaking research community. The most prestigious works are based on LARKS. We think that, the Paolucci et al. matching approach [10] is the most-cited work in this research area. The authors discuss a concept for matching of Web services based on DAML-S. Paolucci et al. define the four levels of exact, plug in, subsume, and fail. These DoMs (Degrees of Match) are based on logic subsumption matching, i.e., the ancestry relationships

between concepts in an ontology. As discussed, a ranking of DoMs is defined. The authors also present the concept of global DoM for two services, which they define as the worst DoM found during matching. Thus, a minimum degree of compatibility regarding service inputs and outputs is achieved. As the related work is very considerable, the following examination will be limited to approaches which have heavily influenced the matchmakers for SAWSDL research community, and further approaches that play an important role in the context of our work. Sivashanmugam et al. [11] present a matchmaking mechanism for WSDL-S which is part of the METEOR-S project. Matching is performed for both functional and QoS requirements. Functional matching can be reduced to semantic matching for operations, inputs, outputs, preconditions, and effects. As a distinctive feature, weights can be assigned to each of the five service elements –i.e., the service requester can rate to which degree each semantic part will be regarded in the overall service matching. The overall semantic matching value is a numerical value in the range of 0 to 1. Syeda-Mahmood et al.[12] WSDL matchmaker combines similarity values based on domain-independent and domain-specific ontologies. Matching is restricted to inputs and outputs. Regarding domain-independent semantics, service element names are captured, tokenized, further processed, and synonyms are detected using WordNet. Domain-specific semantics are based on WSDL-S modelReferences which are similar to the modelReferences defined in SAWSDL. Inferencing is then conducted on the semantic annotations, resulting in one of the possible relationships equivalentClass (0.0), subclassOf (0.5), superClassOf (0.5), and RDFType (0.0). A part from the last-mentioned, these relationships can be traced back to the DoMs defined by Paolucci et al. Afterwards, a numerical value for the semantic relationships is determined; this Value is 1.0 if no relationship could be detected. The overall matching result is the maximum value of the domain-independent or specific similarity values. The matchmaker by Plebani and Pernici, URBE, utilizes linguistic as well as logic information and is applied to WSDL1.1 [13]. The authors argue that Web service descriptions are usually derived from Software components that implement the underlying functionality. In turn, this implies that the contained descriptions will follow some naming convention and thus constitute a meaning full source of information in matchmaking. In URBE, different service abstraction levels are taken into account, namely port types (WSDL1.1's equivalent to interfaces in WSDL2.0), operations, and inputs/outputs.

Klusch et al. provides SAWSDL-MX matchmaker [14] as an extension of MX-matchmakers family originally defined for OWL-S [24]. SAWSDL-MX is a hybrid semantic matchmaker which uses logic-based and text-based similarity information (i.e. cosine based similarity, extended Jacquard-based similarity, intentional loss of information and Jensen-Shannon) to determine the match. There are multiple variants namely, SAWSDL-MX1, SAWSDL-M0+WA (WSDL Analyzer [15]) and

SAWSDL-MX2. Matching is employed on the level of interfaces using a bipartite graph matching algorithm, i.e., similarity values for all combinations of operations from a service request and offer are calculated. Afterwards, the best assignment is calculated. Logic-based matching is conducted for operations and based on the subsumption DoMs by Paolucci et al.[10]. A significant improvement of Paolucci approach was introduced by [44]. The authors used the shortest path algorithm which determines the optimal matching between user query and provider service instead of greedy approach; their contribution comes to reducing the complexity of the Paolucci algorithm.

These systems are adequate when the web service requester and provider use the same ontology and reasoner to determine the relationship between two services. The following paragraph describes the attempts to overcome this limitation.

2. Matching Web Services Using Different Ontologies

The pioneer group of researchers from the University of Georgia [16][17][18] addressed this problem. Cardoso proposes matching without a common ontology commitment. For this purpose, a mapping between concept classes from different ontologies is performed and the geometric distance between the similarities of the domains of the concepts is computed. The mapping is based on syntactic similarity. Like the work by Verma et al. [20], this work was part of the METEOR-S project. Regarding discovery of service, both syntax and semantic-based techniques are deployed. Syntactic similarity measures are deployed for textual service names and service descriptions; semantic-based measures are used to determine the similarity of inputs and outputs. Later, other works have dealt with this problem. Usanavasin [21] proposes an approach to determine the semantic similarity of properties between different ontologies, in order to achieve matchmaking in multi-ontology environment. MOD (Multi Ontology Discovery system) [22], is an algorithm for discovering web services, including the cases where ontology of the request is different from that of the service; it proposed an algorithm for matching of concepts based on [17]. Cardoso [19] presents a technique based on the principle of Teversky to discover web services; the algorithm applied to SAWSDL [5] descriptions but in its multi-ontology version is not too bright. Cardoso paper presents an algorithm to match a semantic Web service request described by SAWSDL against semantic Web service advertisements. The algorithm is novel in three fundamental aspects. First, the similarity among semantic Web service properties, such as inputs and outputs, is evaluated using Teversky's model which is based on concepts (classes), their semantic relationships, and their common and distinguishing features (properties). Second, the algorithm, not only takes into account services' inputs and outputs, but also considers the functionality of services. Finally, the algorithm is able to match a semantic Web service request against advertisements that are annotated with concepts with or without a common

ontological commitment. In other words, it can evaluate the similarity of concepts defined in the context of different ontologies.

3. Ontology Alignment

Aligning ontologies is a crucial task to achieve semantic interoperability in many application domains. Mappings are often generated manually. This process is extremely tedious even if it is facilitated by sophisticated editing tools. Technically we operate on different types of information, names of elements, types of data, representation of the structure of patterns elements, characteristics of the data, etc. Among the several systems that have appeared in recent years a list of non-exhaustive examples includes Prompt [26], ONIONS [27],[28], IF-Map [29], S-Match [30], OLA [31] and more recent works like SOBOM [32], AgrMaker [33], Eff2Match [34], GeRMeSMB [35] and ASMOV [36]. Good and comprehensive state of the art alignment techniques are coverage in [7] and [25].

For example, PROMPT is semi-automatic algorithm for ontology merging and aligning. It begins with the linguistic matchers for initial comparison but guides the user in performing other tasks for which this intervention is required (in choosing the best mappings). ONIONS is a merging approach. It provides articulation rules for resolving terminological heterogeneity and enables knowledge interoperability that will lead to a bridging of the semantic gap between different ontologies. It uses both lexical and graph-based techniques to suggest articulations. The method of finding lexical similarity between concept names uses dictionaries and semantic-indexing techniques based on co-occurrence of words in a text corpus. S-Match is a schema-matching system based on an extensible library of matching generators ranging from lexical methods to SAT solvers. OLA is the only system designed specifically for OWL Lite and which uses a global similarity measure for ontology alignment. Local similarity between entities in the ontologies produces a set of equations which are iteratively solved to provide a set of mappings. ASMOV iteratively computes the similarity by analyzing lexical elements, relational structure, and internal structure. AgrMaker comprises several matching algorithms that can be concept-based or structural. The concept-based matchers support the comparison of strings and the structural matchers include the descendant's similarity inheritance. In AgrMaker the structural similarity depends absolutely on linguistic relationships, which means that correct results are not obtained in ontology concepts which are not linguistically similar, even when the ontologies are very similar in structure.

After this overview of the state of the art, we can conclude that:

- Matchmaking in single ontology context presents shortcomings,
- Semantic interoperability of web services in a multi ontology context is needed,
- Ontology alignment represents the core for

semantic interoperability solution, and,

- Alignment techniques in multi ontologies matchmaking solutions are not used.

The next section presents our ontology alignment proposal to use as a solution for matching Web services.

III. OUR PROPOSAL

Our proposal consists of three essential parts nested one inside the other: a matchmaking algorithm whose basic matching core is a partial alignment mechanism between ontologies, which uses a variety of similarity measures.

A. Matchmaking Algorithm

In this section we present in detail our proposed architecture (see Fig.1 and Fig.2) that contributes to solving different issues discussed in Section 1. This architecture is composed of various modules and their roles are described in depth in this section.

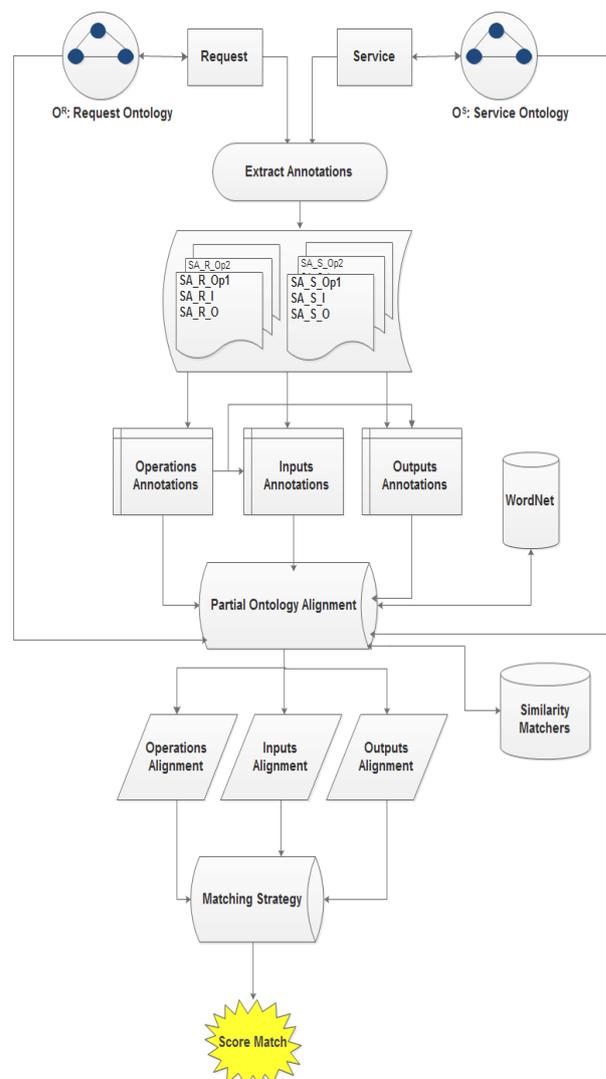


Fig.1. Matchmaking process.

The Service Matchmaking takes two descriptions (request and offer) and returns the degree of match between them. In our work we assume that both service descriptions (request and offer) are expressed in a common unified service model SAWSDL and we assume for SAWSDL-MOM that modelReferences in SAWSDL service offers and requests are pointing to ontological concepts exclusively defined in OWL. The transformation of the service description models into a common service model is out of our work. The first step in the matching of two services is to extract semantic annotations from each description model (SAWSDL). Once the extraction completed, concepts used for annotations are grouped by category; each category (i.e. operations, inputs, outputs) contains two lists (that is, for request and offer) and each couple of lists will be considered as two sub ontologies to align using our partial alignment algorithm. The alignment algorithm uses the basic ontologies as external knowledge in the matching process; each category mapping result will be aggregated into a single score; then the matchmaking strategy determines how to combine the scores to obtain the final score. We use the Hungarian algorithm (Hungarian Method, developed by Harold Kuhn in 1955, is one of the most popular algorithms to solve the assignment problem since it is easy and practical) for the best assignment to determine the 1:1 mappings.

Algo Matchmaking

Inputs : Request, Offer

Outputs : Results

Parameters: Alignment_Cache

{ // If operations are not annotated

For each Operation in Request **do**

For each input in Operation **do**

// SALRI :Semantic Annotations of Request Inputs

SALRI.add(input.modelReference)

For each output in Operation **do**

SALRO.add(output.modelReference)

For each Operation in Offer **do**

For each input in Operation **do**

SALOI.add(input.modelReference)

// SALRI :Semantic Annotations of Offer Inputs

For each output in Operation **do**

{ SALOO.add(output.modelReference)

$I_Match = \text{Partial_alignment}(SALRI, SALOI);$

$O_Match = \text{Partial_alignment}(SALRO, SALOO);$

$Sim_Inputs = \text{agreg}(I_Match);$

$Sim_Outputs = \text{agreg}(O_Match);$

$Result = (Sim_Inputs + Sim_Outputs) / 2;$

Return(Result)}

}

B. Partial Ontology Alignment

In Section II we saw that current service matchmaking algorithms are based on checking the relations between the concepts that appear in the common field of semantic service descriptions. If the concepts being compared are defined in different ontologies then semantic alignments must be considered instead of obtaining a fail match. In

this work we are concerned with ontology alignment techniques, and the use of alignments. The nature of the existing ontology matching tools is different from our matching mechanism. We assume in our matching that:

- Partial match considers only some elements of the ontology,
- Relationship cardinalities between entities are one-to-one.

A more comprehensive view of the matching problem of two Web service descriptions may be considered as a multi alignment problem; we restrict our study to the problem of matching of two ontologies.

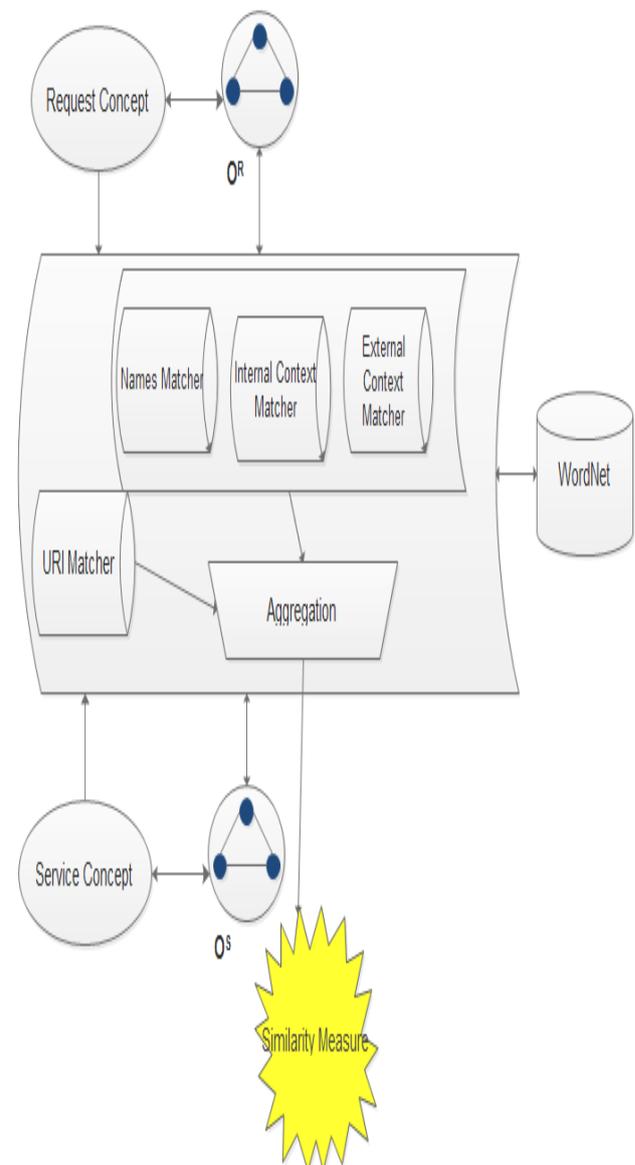


Fig.2. Concepts Matching Process.

Three procedures have been designed for the mapping process: (1) concepts selection; (2) similarity calculation and (3) best assignment. Concepts selection is designed to minimize the number of concepts to be mapped between two ontologies such that efficient online

ontology mapping can be achieved. It is especially important for mapping large ontologies. Similarity calculation estimates the similarity between two concepts by looking at their names, internal and external contexts.

```

Algo Compute_Overall_Sim
Inputs :CR, CS: two concepts
Outputs:result
Parameters:WName,WProp,WFather
{
  Sim_Name=Compute_Sim_Name(CR.name,CS.name);
  Sim_Prop=Compute_Sim_Prop(CR.prop, CS.prop);
  Sim_Vois=Compute_Sim_Father(CR, CS) ;
  Result=WName * Sim_Name+WProp * Sim_Prop+WFather
  *SimFather;
  Return(Result)
}

```

C. Similarity Between two Concepts across Ontology

One of the most important parts of a matching process is the similarity function because it “decides” how two concepts are similar. For our approach we need several simple similarity functions which will be defined in the following.

To align the concept CR and the concept CS belonging to two different ontologies, we start by comparing their URIs. If they are identical then the mapping is perfect, else we calculate an overall similarity measure, which is an aggregation of several similarities, in particular, the similarity of names, and the similarity of the internal and external contexts. Fig.2 presents the concepts matching process. The detail of the individual matchers, namely *Names Matcher*, *Internal Context Matcher* and *External Context Matcher*, invoked by the *Compute_Overall_Sim* algorithm is presented below.

```

//Computing the similarity between two concepts
//*****
Algo Compute_Similarity
Inputs :CR :Request Concept ;
          CS:Service Concept ;
Outputs : Result :Overall similarity;
{
  If (CR, CS) in Alignment_Cache
  return(result from Alignment_Cache)
  Else
  {
    Sim_URI=Compute_sim_uri(CR,CS) ;
    If(Sim_URI=1)
    return(Result=Sim_uri)
    Else
    Result=Compute_Overall_Sim(CR,CS);
  }
  return(Result)
}

```

```

// PartialAlignment
//*****
Algo partial_alignment
Inputs :LCS,LCC: two lists of concepts
Outputs :Result:one to one mappings
Parameters: ontologies, wordnet,basic matchers
{
  For each cs in LCS do
  For each cc in LCC do
  Sim(i,j)=Compute_Similarity(cs,cc);
  Result=optimal_assignment(Sim);
}

```

1. Names Matcher (names similarity)

$$Sim_{Name}(T_{CR}, T_{CS}) = \frac{Sim_{Syn}(T_{CR}, T_{CS}) + Sim_{Ling}(T_{CR}, T_{CS})}{2} \quad (1)$$

where:

$Sim_{Syn}(T_{CR}, T_{CS})$: is the syntactic similarity between the terms T_{CR} and T_{CS} respectively associated with the concepts CR and CS.

$Sim_{Ling}(T_{CR}, T_{CS})$: is the linguistics similarity between the terms T_{CR} and T_{CS} associated with the concepts CR and CS respectively.

If ($|T_{CR}| > 1$) or ($|T_{CS}| > 1$) then we compute a matrix M where:

$M(i, j) = Sim_{Name}(T_{CR}^i, T_{CS}^j)$ from which we obtain the vector:

$$V = \text{BestAssignment}(M)$$

$$Sim_{Name}(T_{CR}, T_{CS}) = \sum \frac{V(i)}{\text{Max}(n,m)} \quad (2)$$

$$n = |T_{CR}| \text{ and } m = |T_{CS}|$$

The Syntactic similarity Sim_{Syn} is called Levenshtein similarity, which measures the similarity of two strings on a scale from 0 to 1 based on Levenshtein’s edit distance, ed [37]. It is given by:

$$Sim_{Syn} = \frac{\text{Max}(|T_{CR}|, |T_{CS}|) - \text{ed}(T_{CR}, T_{CS})}{\text{Min}(|T_{CR}|, |T_{CS}|) + \text{ed}(T_{CR}, T_{CS})} \quad (3)$$

Where:

$|T_{CR}|$: is the number of characters in the term T_{CR} of the concept CR; likewise for $|T_{CS}|$.

$\text{ed}(T_{CR}, T_{CS})$: is the edit distance that represents the number of transformations (addition, suppression, modification) needed to find T_{CR} from T_{CS} .

Our measure is an improvement of work [7]. For example, if T_{CR} ='Professor' and T_{CS} ='Assistant Professor' their measure is limited to 0, but our measure returns 0.47, a significant value.

The Linguistic similarity Sim_{Ling} is based on WordNet, which is an online lexical database of English, developed under the guidance of Miller at Princeton University [38]. Here, a set of cognitive synonyms called synsets, each representing a different concept, are formed by grouping the nouns, verbs, adjectives and adverbs. Synsets are created by using conceptual semantic and lexical relations. WordNet can also be seen as ontology for natural language terms. Our linguistic similarity measure (Sim_{Ling}) is formulated as follows:

$$Sim_{Ling}(CR, CS) = \frac{|Syn(T_{CR}) \cap Syn(T_{CS})|}{|Syn(T_{CR}) \cup Syn(T_{CS})|} \quad (4)$$

Where:

$Syn(T_{CR})$: All synsets of the term T_{CR} ,

$Syn(T_{CS})$: All synsets of the term T_{CS} .

2. Internal Context Matcher

Internal context is composed by the internal properties of a concept. A concept may have one or more properties. Similar to a concept, a property also has a name and a description. In addition, it contains domain, range and cardinality. To compute the property similarity, whereas we can exploit all or a part of this information of the two properties, in our case only the names and ranges are

$$Sim_{Range}(P_i, P_j) = \begin{cases} 1 & \text{if Range}(P_i) = \text{Range}(P_j) \\ 0.9 & \text{if Range}(P_i) = \text{AnyType} \text{ and Range}(P_j) = \text{String} \\ 1 & \text{if Range}(P_i) = \text{Integer} \text{ and Range}(P_j) = \text{Float} \\ 1 & \text{if Range}(P_i) = \text{Integer} \text{ and Range}(P_j) = \text{Double} \\ \frac{2}{3} & \text{if Range}(P_i) = \text{Float} \text{ and Range}(P_j) = \text{Integer} \\ 1 & \text{if Range}(P_i) = \text{Float} \text{ and Range}(P_j) = \text{Double} \\ \frac{1}{3} & \text{if Range}(P_i) = \text{Double} \text{ and Range}(P_j) = \text{Integer} \\ \frac{1}{2} & \text{if Range}(P_i) = \text{Double} \text{ and Range}(P_j) = \text{Float} \\ \dots & \dots \\ 0 & \text{Otherwise} \end{cases} \quad (7)$$

3. External Context Matcher

Structural techniques consist in exploiting the structure of the ontologies to be compared, often represented as graphs. Similarity between two entities from two ontologies can be based on the position of entities in their hierarchies [30]. These techniques use various heuristics and are based on the following hypothesis: if two entities are similar in two ontologies, their neighbors are also somehow [25]. This remark can be used in several ways. Examples of criteria for deciding if two entities are similar are cited below:

C1: two concepts are similar if their "super-concepts" ("Fathers") are similar;

matched. Initially we compute the matrix of similarities between the properties of two concepts CR and CS, then we apply the Hungarian algorithm to obtain similarity vector V_{Sim} for two lists of properties and finally, we get their similarity:

$$Sim_{Prop}(CR, CS) = \frac{1}{\text{Max}(n,m)} \sum V_{Sim}(i) \quad (5)$$

where:

n: the number of properties of CR,
m: the number of properties of CS, and
 $V_{Sim} = \text{BestAssignment}(Sim_{Prop})$.

The similarity vector obtained by Hungarian algorithm is applied to similarity matrix Sim_P .

$Sim_P(P_i, P_j)$ is the similarity between the property P_i of concept CR and the property P_j of the concept CS, calculated according to the equation:

$$Sim_P(P_i, P_j) = W_{Name} * Sim_{Name}(P_i, P_j) + W_{Range} * Sim_{Range}(P_i, P_j) \quad (6)$$

Where: $W_{Name} + W_{Range} = 1$

where $Sim_{Name}(P_i, P_j)$ is the similarity of names of the two properties. To find the similarity of the names, we use the names similarity algorithm eq (1) and (2).

The property range similarity is calculated as follows:

C2: two concepts are similar if their "sub-concepts" ("Sons/Daughters") are similar;

C3: two concepts are similar if their "Siblings/Neighbors" are similar.

Naturally, an approach can combine several criteria like those above. We suggest an approach in calculating the structural similarity between the entities of two ontologies where we are inspired from the works of Abolhassani [39] and Fellah [40].

The external context of a concept is usually known by its Super, Sub and Sibling (3S) concepts in the respective ontology. A concept may or may not have sub or sibling concepts but it always has some super-concepts. This

means that while identifying contextual similarity between two concepts, the similarity between their respective super concepts should be considered only.

3.1. Intuition and Theoretical Basis

By the C1, C2, and C3 criteria, the similarity between a concept $CR \in OR$ and $CS \in OS$ depends on the similarity between their contexts. In other words, if the context of a concept CR denoted Super (CR) is similar to the context of a concept CS denoted Super (CS) then CR and CS are similar in some sense.

Definition (The Relative External Context):

Relative External Context (REC) of a couple (CR, CS) (with $CR \in OR$ and $CS \in OS$) is defined by the pairs (eR, eS) as the following conditions are satisfied:

$$\forall eR \in OR \text{ and } \forall eS \in OS \\ REC(CR, CS) = \{(eR, eS) / eR \in Super(CR) \text{ and } \\ eS \in Super(CS) \text{ and } BestAssignment(eR, eS)\}$$

$BestAssignment(eR, eS)$: means that eR is aligned with eS .

The similarity between two concepts CR and CS depends on the link between their external contexts $(eR, eS) \in REC(CR, CS)$ and also on the strength of the link of CR and CS with their corresponding super concepts eR and eS . We quantify this measure by taking into account the similarity between contexts of two concepts and the connectivity between a concept and its context. The connectivity between CR and eR is semantic distance between them. It is difficult to choose between the semantic similarity measures, and especially which would provide the most relevant results in our case. We rely here on the measure of Wu and Palmer [41], and it is chosen because of its implementation simplicity. Wu and Palmer similarity measure Sim_{WP} is expressed as follows:

$$Sim_{WP}(c1, c2) = \frac{\text{depth}(LCA(c1, c2))}{\text{depth}(c1) + \text{depth}(c2)} \quad (8)$$

Where $LCA(c1, c2)$ is the least common ancestor (Lowest Common Ancestors of $c1$ and $c2$) and $\text{depth}(c1)$ is the path length between $c1$ and the root passing through $LCA(c1, c2)$.

We start by computing the matrix of similarity M between Fathers (i.e, Super(CR) and Super(CS)) then we choose the best couples;

$$V_{Sim} = BestAssignment(M).$$

We calculate the similarity for fathers across ontologies as follows:

$$Sim_{Father}(CR, CS) = \frac{(\sum (Ql * (1 - |Sim_{WP}(CR, eR) - Sim_{WP}(CS, eS)|)))}{MaxF} \quad (9)$$

where:

Ql : represents the similarity between pairs of fathers $(eR, eS) \in REC(CR, CS)$

$$MaxF = \text{Max}(|Fathers(CR)|, |Fathers(CS)|)$$

It easy to verify that equation (09) satisfies the properties of a similarity measure:

- 1) $C \in \text{Ontology}, Sim_{Father}(C, C) = 1$
- 2) $x, y \in \text{Ontology}, Sim_{Father}(x, y) = Sim_{Father}(y, x)$
- 3) $0 \leq Sim_{Father} \leq 1$

Finally the Overall similarity is the aggregation of all similarities (i.e. names, properties and father's similarities) into a global similarity measure given by the following equation:

$$Sim_{Overall}(CR, CS) = W_{Name} * Sim_{Name}(CR, CS) + \\ W_{Prop} * Sim_{Prop}(CR, CS) + W_{Father} * \\ Sim_{Father}(CR, CS) \quad (10)$$

$$W_{Name} + W_{Prop} + W_{Father} = 1$$

If Proprieties $(CR) = \emptyset \vee$ proprieties $(CS) = \emptyset$
Then $W_{Prop} = 0$.

IV. EXPERIMENTAL EVALUATION

Our experiment is broken down into three parts: the first is devoted to the evaluation of the partial alignment algorithm, the second for evaluating our matchmaking algorithm with SAWSDL-TC [42] and the third evaluation is a way to a new test base for multi matchmaking, which allows us to proceed to an evaluation in a multi ontologies environment.

A. Partial alignment evaluation

The nature of the existing ontology matching tools is different from our matching mechanism; therefore an equitable comparison of the performance is difficult to achieve. Our matching mechanism tries to find correspondences for the concepts appearing in the query while existing alignment tools search the mappings for all concepts. Since it is difficult to make fair comparisons, experiences must be carefully designed. Evaluation results will be presented in the last part of this section. Our goal is to design a partial and efficient alignment algorithm for a dynamic environment. Efficacy may be measured by the speed and precision of alignment results. The algorithm's speed is measured by the time used to perform ontology mapping. To measure the precision of partial mappings between two ontologies, manual mapping references are needed. We conducted a series of tests using the base Benchmark of Ontology Alignment Evaluation Initiative OAEI-2011 Campaign for the EON competition [43]. The ontology base consists of a set of bibliographic references. It is a leaner version with the number of ontological entities compared with the real

ontology. Each test case of the basic benchmark highlights a feature of the second ontology aligned with the test database. The basic objective of these tests is to assume all aspects that exist in the OWL ontology and then see what impact that could have on the alignment results. We executed 5 test cases; each test considers a set of a number of concepts (for example 2, 4, 6, 8 and 10

concepts) from 101 ontology and another set of concepts in the target ontology variant from 101,103, 104, 202, 230, 301 and 304.

In the following the different tests and their results are briefly described. An overview of the complete results is shown in Fig.3.

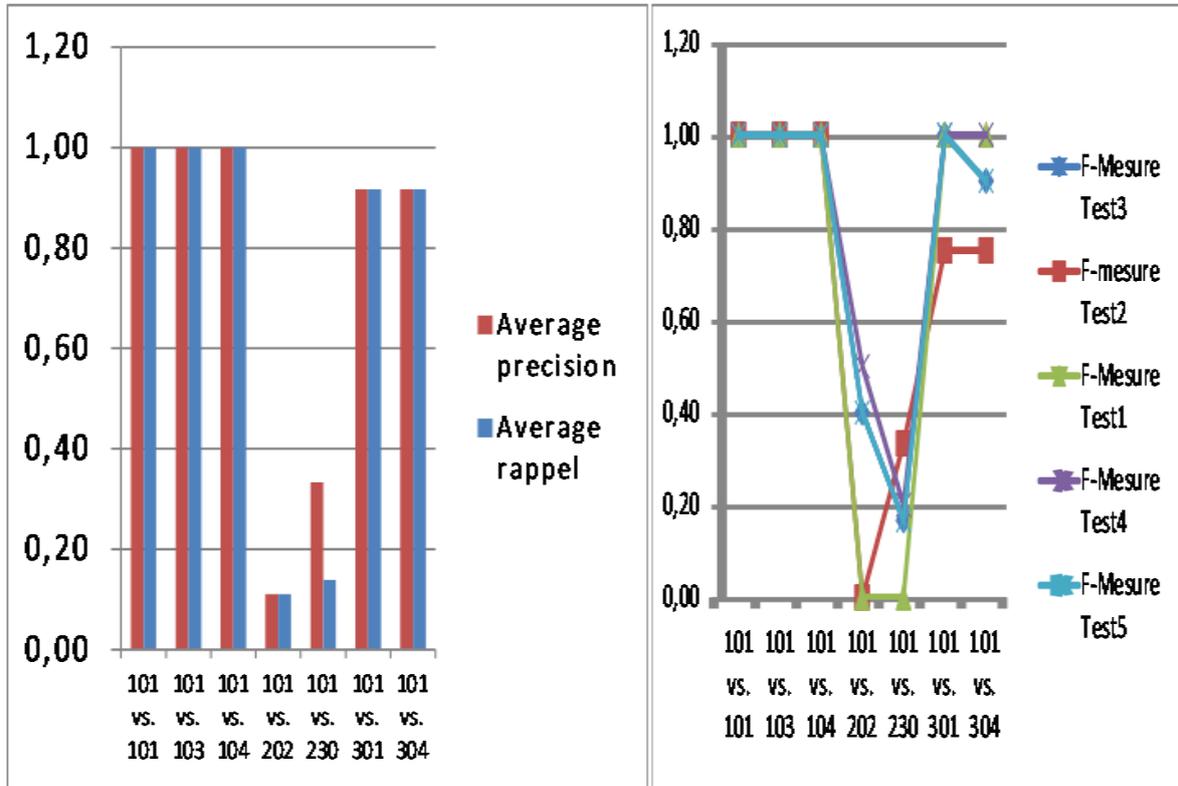


Fig.3. Partial Alignment Tests.

The alignment produced at each test is compared with the reference alignment. Thus, the quality measurement values of alignment (precision, recall, and F-measure) are determined. The best results precision values are obtained when ontology structures are similar (or identical), i.e. the families of tests10x and 30x. Thus, our method obtains precision values for these tests which are equal to 1.00. This is due to the fact that our approach explores entities structures to align through the structural similarity.

B. SAWSDL Matchmaking evaluation

We conducted a series of tests using the base Benchmark SAWSDL-TC3. Currently it is the only standard test collection for SAWSDL services matchmaking and must be used for comparison with other approaches. SAWSDL-TC3 consists of a collection semantically annotated WSDL 1.1 based Web services, which cover differing domains. The Benchmark includes queries and offers. Requests and offers are both encoded using SAWSDL. The base designers added a binary and graded relevance set for each query, which can be used to compute evaluation metrics. In SAWSDL-TC, semantic annotations exist solely at message parameter level.

SAWSDL-MOM incorporates information from the operation, and message parameter levels of SAWSDL. In the following, we will present the most important of our experience. The Multi matchmaking approach presented in Section 3 has been implemented in SAWSDL-MOM (Multi Ontology Matcher using Partial Ontology Alignment) using Pellet as reasoner and JWNL as interface to WordNet. As test data collection, SAWSDL-TC3 has been adopted. In accordance with the procedure in the S3 Contest, we evaluated the IR metrics automatically computed by SAWSDL-MX V2.0 like SME, namely Average Precision (AP) and Average Query Response Time (AQRT) in the evaluation results. We have compared our results with SAWSDL-MX1 variants which are:

- SAWSDL-M0: Logic Based Matcher,
- SAWSDL-M1: Hybrid Matcher using Loss of Information Similarity,
- SAWSDL-M2: Hybrid Matcher using Extended Jaccard Similarity,
- SAWSDL-M3: Hybrid Matcher using Cosine Similarity,

SAWSDL-M4: Hybrid Matcher using Jensen Shannon Similarity.

The following table provides a comparison of multiple matchmakers regarding the criterion of performance:

Table 1. Results of Matchmaker Comparisons.

Matchmaker	1 Request 20 Services		2 Requests 20 Services		4 Requests 20 Services	
	AP	AQRT	AP	AQRT	AP	AQRT
SAWSDL-M0	0.500	2255	0.667	3698	0.75	1873
SAWSDL-M1	0.883	1502	0.922	3068	0.941	1866
SAWSDL-M2	0.840	1512	0.894	3604	0.942	1892
SAWSDL-M3	0.849	1770	0.899	3814	0.925	1632
SAWSDL-M4	0.840	1925	0.894	3416	0.920	1542
SAWSDL-MOM	0.905	5551	0.937	10202	0.952	12088

Our matchmaker complements and furthers already existing work by combining accepted techniques and new ideas achieving good results regarding IR metrics such as recall and precision and provided good AP values better than any matchmaker variants of SAWSDL-MX1.

C. Towards an Extended SAWSDL-TC

The principle of our extension is inspired from the basic test used for evaluating ontology alignment algorithms already presented. The idea is to process alterations in the ontology (ontology that annotates a web service) and will be considered as basic ontology, generating new ontologies thing that affects the services annotated by the basic ontology and will have new web services. All of the well generated Web services will build our new basic tests for multi matching. Given the importance of the work and in order to clarify ideas, we consider a simplified choice from SAWSDL-TC as follow:

The request

- BookPriceService

And the services:

- NovelPriceService
- Short-storyAuthorbook-typeService

Example Basic ontology Books.owl and web service BookPriceService annotated with the concept #Book belonging to the ontology Books.owl. We proceed to the alterations:

Alteration 1:

Construction of a new ontology books1.owl exactly similar to books.owl

Alteration 2:

Construction of a new ontology books2.owl exactly similar to books.owl renaming the concept #Book by #Text_Book.

These changes generate a second web service BookPrice1 which is annotated with the concept

#Text_Book.

Alteration 3:

Construction of a new ontology books3.owl exactly similar to books.owl renaming the concept #Book with #hththf (any string).

And so on, the changes are done in the test base of EON competition [43]. We have used Protégé 4.0 to create our OWL ontologies and SyncRO Soft, S. R. L. Oxygen XML Editor Version 17.1 to edit and annotate Web services. So we can express all possible heterogeneities between ontologies to test the efficiency of matchmaking tools. Therefore the three alterations of book.owl generate ontology for each service of test base SAWSDL-TC, three other services annotated with the variants #Book concept, and so, the base will be enriched by an exhaustive number of study cases to enable us to make a sound judgment of the matchmakers. Experiments performed on above changes clearly show the efficiency of our ideas and the use of ontology alignment relative to other matchmaking tools centered on the use of a common ontology. One of our projects in the near future is the development of a test database for SAWSDL rich enough to support all aspects of interoperability of Web services in a multi-ontology environment.

V. CONCLUSION

In this paper, we proposed a multi-ontology matching approach to service matchmaking. We allow the matching of semantic Web services with the use of a common ontology or different ontologies. This aspect is important since it is not pragmatic to assume that Web services will always be defined by a single ontology. In some cases, similar services may be defined by different ontologies. We proposed an architecture that has a partial semantic alignment as a core component. We provided ideas and developments towards the construction of a service matchmaking framework in which semantic alignment mechanisms are purposefully integrated into. We evaluated our matchmaker for SAWSDL, but the SAWSDL-MOM algorithm is general; therefore it can be used for web services using different semantic Web

service description languages such as OWL-S, and WSML. However, the partial ontology alignment algorithm can be applied to other areas that require interoperability. The matching process can be easily extended to include non-functional capabilities of services.

The results of the experimental evaluation of SAWSDL-MOM provide strong evidence for the assertion that the introduction of ontology alignment techniques for SAWSDL Web services matchmaking can significantly improve results, achieve better interoperability and open the way to real scalability. In the future we intend to extend this work to incorporate the following:

- Improving the similarity measure,
- Adding optimizations to the alignment algorithms to make it real time,
- Adding the matching of non-functional parameters,
- Applying our ideas to other Web services representation models such as OWL-S,
- Designing a test collection that supports web services annotated with multiple ontologies.

REFERENCES

- [1] Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., & Orchard, D. (2013). Web services architecture, w3c working group note, 2004. URL: <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>.
- [2] Mohebbi, K., Ibrahim, S., Khezrian, M., Munusamy, K., & Tabatabaei, S. G. H. (2010). A comparative evaluation of semantic web service discovery approaches. In Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services (pp. 33-39). ACM.
- [3] Weise, T., Blake, M. B. & Bleul, S. (2014). Semantic Web Service Composition: The Web Service Challenge Perspective. In Web Services Foundations (pp. 161-187). Springer New York.
- [4] Sure., M.E.a.Y. (2004). Ontology mapping – an integrated approach. In Proceedings of the First European Semantic Web Symposium. Heraklion, Greece: Lecture Notes in Computer Science.
- [5] Farrell, J. & Lausen, H. (2007). Semantic annotations for WSDL and XML schema. W3C recommendation.
- [6] Berners-Lee, T., J. Hendler, and O. Lassila. (2001). The Semantic Web. A new form of Web content that is meaningful to computers will leash a revolution of new possibilities. *Scientific American*, 284(5): 34–43.
- [7] Euzenat, J. & Shvaiko, P. (2007). *Ontology Matching*. Springer, Heidelberg.
- [8] Kalfoglou, Y. & Schorlemmer, M. (2003). Ontology mapping: the state of the art. *The knowledge engineering review*, 18(01), 1-31.
- [9] Katia P. Sycara, Matthias Klusch, Seth Widoff, and Jianguo Lu. *Dynamic Service Matchmaking among Agents in Open Information Environments*. SIGMOD Record, 28(1):47–53, 1999.
- [10] Paolucci, M., Kawamura, T., Payne, T. R. & Sycara, K. (2002). Semantic matching of web services capabilities. In *The Semantic Web—ISWC 2002* (pp. 333-347). Springer Berlin Heidelberg.
- [11] Sivashanmugam, K., Verma, K., Sheth, A. and Miller, J. A. Adding Semantics to Web Services Standards. In *International Conference on Web Services (ICWS2003)*, pages 395–401. CSREA Press, 2003.
- [12] Syeda-Mahmood, T. F., Gauri Shah, Rama Akkiraju, Anca-Andreea Ivan and Richard Goodwin. *Searching Service Repositories by Combining Semantic and Ontological Matching*. In *2005 IEEE International Conference on Web Services (ICWS 2005)*, pages 13–20. IEEE Computer Society, Washington, DC, USA, 2005.
- [13] Plebani, P. & Pernici, B. (2009). URBE: Web service retrieval based on similarity evaluation. *Knowledge and Data Engineering, IEEE Transactions on*, 21(11), 1629-1642.
- [14] Klusch, M., Kapahnke, P. & Zinnikus, I. *Hybrid adaptive web service selection with SAWSDL-MX and WSDL-analyzer*. 2009. Heraklion, Crete, Greece: Springer Verlag.
- [15] Zinnikus, I., Rupp, H. J. & Fischer, K. (2006). Detecting similarities between web service interfaces: The WSDL analyzer. In *Interoperability for Enterprise Software and Applications: Proceedings of the Workshops and the Doctoral Symposium of the Second IFAC/IFIP I-ESA International Conference: EI2N, WSI, IS-TSPQ 2006* (pp. 145-156). ISTE.
- [16] Cardoso, J. & Sheth, A. (2003). Semantic e-workflow composition. *Journal of Intelligent Information Systems*, 21(3), 191-225.
- [17] Oundhakar, S., Verma, K., Sivashanmugam, K., Sheth, A. P. & Miller, J. A. (2005). Discovery of web services in a multi-ontology and federated registry environment. *International Journal of Web Services Research*, 2(3), 1-32.
- [18] Cardoso, J. (2006, September). Discovering semantic web services with and without a common ontology commitment. In *Services Computing Workshops, 2006. SCW'06. IEEE* (pp. 183-190). IEEE.
- [19] Cardoso, J., Miller, J. A. & Emami, S. (2008). Web services discovery utilizing semantically annotated WSDL. In *Reasoning Web* (pp. 240-268). Springer Berlin Heidelberg.
- [20] Verma, K., Sheth, A., Oundhakar, S., Sivashanmugam, K. & Miller, J. (2007). Allowing the use of Multiple Ontologies for Discovery of Web Services in Federated Registry Environment. Department of Computer Science, University of Georgia, Athens, Georgia. Technical Report# UGA-CS-LSDIS-TR-07-011, 1-27.
- [21] Usanavasin, S., Takada, S. & Doi, N. (2005). Semantic web services discovery in multi-ontology environment. In *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops* (pp. 59-68). Springer Berlin Heidelberg.
- [22] Le DuyNgan, T. M. H. & Goh, A. E. S. (2006, September). MOD-A Multi-Ontology Discovery System. In *1st International Workshop on Semantic Matchmaking and Resource Retrieval* (p. 3).
- [23] Klusch, M. (2008). Semantic web service coordination. In *CASCOM: Intelligent Service Coordination in the Semantic Web* (pp. 59-104). Birkhäuser Basel.
- [24] Klusch, M. & Kapahnke, P. (2009, October). OWLS-MX3: an adaptive hybrid semantic service matchmaker for OWL-S. In *Proceedings of 3rd International Workshop on Semantic Matchmaking and Resource Retrieval (SMR2)*, USA.
- [25] Euzenat J., et al. (2004). D2.2.3: State of the art on ontology alignment. Technical report, NoE Knowledge Web project deliverable, 2004. <http://knowledgeweb.semanticweb.org/>.
- [26] Noy, N.F. & Musen, M.A. (2003). The prompt suite: interactive tools for ontology merging and mapping. *Int. J. Hum.-Comput. Stud.* 59(6), 983–1024.

- [27] Gangemi A., Steve G. & Giacomelli F. (2007). ONIONS: An Ontological Methodology for Taxonomic Knowledge Integration”, Repart of Informatica Medica, Istituto Tecnologie Biomediche, CNR, Roma, Italy
- [28] McGuinness, D.L., Fikes, R., Rice, J., & Wilder, S. (2000). An environment for merging and testing large ontologies. In: Proceeding of KR, pp. 483–493.
- [29] Kalfoglou, Y. & Schorlemmer, M. (2003). IF-Map: An ontology-mapping method based on information- flow theory. In Journal on data semantics I (pp. 98- 127). Springer Berlin Heidelberg.
- [30] Giunchiglia, F. & Shvaiko, P. (2003). Semantic matching. The Knowledge Engineering Review 18(3), 265–280.
- [31] Valtchev, P. Euzenat, J., Loup, D. & Touzani, M. (2004). Ontology alignment with OLA. In Proc. 3rd ISWC2004 workshop on Evaluation of Ontology- based tools (EON) (pp. 59-68).
- [32] Xu, P., Wang, Y., Cheng, L. & Zang, T. Alignment Results of SOBOM for OAEI 2010. In Proceeding of 5th International Workshop on Ontology Matching (OM 2010), Shanghai, China, 7–11 November 2010.
- [33] Cruz, I.F., Antonelli, F.P. & Stroe, C. Agreement Maker Efficient Matching for Large Real-World Schemas and Ontologies. In Proceeding of International Conference on Very Large Databases, Lyon, France, September 2009; pp. 1586–1589.
- [34] Chua, W.W.K. & Kim, J.J. Eff2Match Results for OAEI 2010. (2010). In Proceeding of 5th International Workshop on Ontology Matching (OM 2010), Shanghai, China, 7–11 November 2010.
- [35] Quix, C., Gal, A., Sagi, T. & Kensch, D. An Integrated Matching System: GeRoMeSuite and SMB-Results for OAEI 2010. In Proceeding of 5th International Workshop on Ontology Matching (OM 2010), Shanghai, China, 7–11 November 2010.
- [36] Jean-Mary, Y., Shironoshita, E.P. & Kabuka, M. Ontology matching with semantic verification. Web Semant. Sci. Serv. Agents World Wide Web 2009, 7, 235–251.
- [37] Levenshtein, I.V. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Cybernetics and Control Theory.
- [38] Miller. G.A. (1995). WordNet: A lexical Database for English. Comm. ACM, Vol. 38, No. 11, pp. 39-41.
- [39] Abolhassani H., B.B. Hariri & S. H. Haeri. (2006). On Ontology Alignment Experiments, Webology, Volume 3, Number 3.
- [40] Fellah, A., Malki, M. & Zahaf, A. (2008). Alignement des ontologies: utilisation de WordNet et une nouvelle mesure structurelle. In Conférence en Recherche d’Information et Applications, CORIA, 2008. pp.401-408.
- [41] Wu Z. & Palmer M. (1994). Verb semantics and lexical selection. In proc. of the 32nd Annual Meeting of Computational Linguistics, Las Cruces, 1994, pp. 133-138.
- [42] SAWSDL-TC: SAWSDL Service Retrieval Test Collection. Latest version SAWSDL-TC3.0 (SAWSDL-TC3) published on September 22, 2009, at semwebcentral: <http://projects.semwebcentral.org/projects/sawSDL-tc/>.
- [43] Ontology Alignment Evaluation Initiative Benchmark Test Library 2011. Retrieved July 7,2015 from <http://oaei.ontologymatching.org/2011/benchmarks>.
- [44] Khater, M. & Malki, M.(2014). "Improving the Performance of Semantic Web Services Discovery: Shortest Path based Approach", IJITCS, vol.6, no.7, pp.32-39, 2014. DOI: 10.5815/ijitcs.2014.07.05

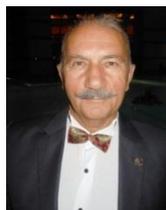
Authors’ Profiles



Aissa FELLAH he is currently Assistant Professor at Computer Science Department in Tahar Moulay University of Saida. After 10+ years in IT industry, he is pursuing PhD program in Computer Science at the Computer Science Department in Djillali Liabes University of Sidi Bel-Abbes, Algeria. He is a member of EEDIS lab in UDL-SBA and a member of LabRI-SBA Laboratory. His academic interests include semantic Web services and ontology matching.



Mimoun MALKI is currently a full professor at the Ecole Supérieure en Informatique de Sidi Bel-Abbes. He is the head of the LabRI-SBA Laboratory. His research interests include Databases, Information Systems Interoperability, Ontology Engineering, Semantic Web Services, Linked Data Services, Web Reengineering, Enterprise Mashup and Cloud Computing. He has published papers in well-known journals (IEEE Transactions on Knowledge and Data Engineering, World Wide Web Springer, etc.)



Atilla ELÇI is full professor and chairman of Electrical and Electronics Engineering at Aksaray University, Turkey. He served as full professor of computer engineering, the founding director of the Graduate School of Science and Technology, and the dean of Engineering Faculty at Toros University, Mersin, Turkey (July 2010 - June 2011) and at various other universities. He has organized or served in the committees of numerous international conferences. He has published over a hundred journals and conference papers; co-authored Composition of OWL-S based Atomic Processes - Towards a Semantic-based Workflow Model (LAP Lambert, 2011); edited Semantic Agent Systems (Springer, 2011), Theory and Practice of Cryptography Solutions for Secure Information Systems (IGI, 2013) and Handbook of Applied Learning Theory and Design in Modern Education (IGI, 2016); proceedings SIN 2007, 2009 - 2015 by ACM, ESAS 2006-15 by IEEE CS, and IJRCS 2009; and various special issues and book chapters. He was the program chair for the 36th IEEE CS COMPSAC (2012) and standing committee member since then. He is an associate editor of Expert Systems: The Journal of Knowledge Engineering and editorial board member of several other journals. He obtained B.Sc. in Computer/Control Engineering at METU, Ankara, Turkey (1970), M.Sc. & Ph.D. in Computer Sciences at Purdue University, USA (1973, 1975).

How to cite this paper: Aissa FELLAH, Mimoun Malki, Atilla ELÇI, "Web Services Matchmaking Based on a Partial Ontology Alignment", International Journal of Information Technology and Computer Science(IJITCS), Vol.8, No.6, pp.9-20, 2016. DOI: 10.5815/ijitcs.2016.06.02