A Multimedia Web Service Matchmaker

Sid Midouni[1,2], Youssef Amghar[1], and Azeddine Chikh[2]
[1]Université de Lyon, CNRS INSA-Lyon, France
[2]Département d'informatique, Université Abou Bekr Belkaid-Tlemcen, Algérie

Abstract: *The full service approach for composing (MaaS) Multimedia as a Service in multimedia data retrieving, which we have proposed in a previous work, is based on a four phases process: description; matching; clustering; and restitution. In this article, we show how MaaS services are matched to meet user needs. Our matching algorithm consists of two steps: (1) the domain matching step is based on the calculation of similarity degrees between the domain description of MaaS services and user queries; (2) the multimedia matching step compares the multimedia description of MaaS services with user queries. The multimedia description is defined as a SPARQL Protocol and RDF Query Language(SPARQL) query over multimedia ontology. An experimentation in a medical domain allowed to evaluate the solution. The results indicate that using both domain and multimedia matching considerably improve the performance of multimedia data retrieving systems.*

1. Introduction

Accessing multimedia data in distributed systems poses new challenges due to many system parameters: volume, diversity of interfaces, representation format, location, etc. In addition, the growing needs of users and applications to incorporate semantics in the information retrieval cause also new issues. Current solutions, which are mainly based on integration or interoperability, are often unsatisfactory considering the multimedia content flows in a holistic way. In our past research, we have proposed a new full service approach [13]. It overcomes some missing issues in accessing and searching multimedia data in the context of distributed and heterogeneous systems. A new pattern of services was defined: multimedia web services Multimedia as a Service (MaaS). MaaS is a specific data web service that accesses to multimedia data. It gives a relevant answer to the user, based on a four phase's process: description, matching, clustering and restitution. This paper, which is a continuation of our previous work, gives an overview of MaaSs description and explains in detail how MaaSs are matched with user needs. Matching is then one of our study objects. Different approaches for matching semantic web services have been developed in the literature [3]. In order to address the problem of matching between MaaSs and user needs, we propose a new matching mechanism based on the MaaS description approach. This matching mechanism is twofold: domain and multimedia matching. The multimedia matching is performed if and only if the domain matching has succeeded. The domain matching adopts an algorithm based on the calculation of similarity degree between semantic concepts annotating MaaSs in one side and those annotating the query in another side. The multimedia matching adopts an algorithm for comparing multimedia description of MaaSs and query. The multimedia description is defined as a SPARQL Protocol and RDF Query Language (SPARQL) query over multimedia ontology.

The remaining of this paper is organized as follows: section 2 presents a state of the art on semantic matching approaches. section 3 defines formallyour multimedia web services MaaSs. section 4 describes briefly the most important multimedia ontologies. Then, we give in section 5 an overview of our MaaSs description approach as well as our matching algorithm. Section 6 presents the experiments achieved to validate the proposed solution. The last section is devoted to the conclusion and future works.

2. Related Work

Service matching is the act of finding relevant services for a user request. Web service matching seems to be similar to the matching problems in other areas, such as database matching, text matching and software pattern matching. However, web service matching is still different requiring specific techniques. Indeed, the matching approaches depend on the service description parts to match. Existing approaches focus either on service process; service profile (functional, non-functional, etc.,); or both of them. Three categories of approaches of web service matching exist and depend of the way to perform the matching that can be logic-based or not or both.

The logic-based matching approaches use ontological concepts and logical rules. Matching degrees are defined differently depending on semantics of matched description elements. There are mainly three matching approaches [2]:

1. IO-matching determined from semantic data service parameters: Inputs (I) and Outputs (O).
2. PE-matching determined from matching on Pre-conditions (P) and Effects (E) of services and queries.
3. IOPE-matching: determined from matching semantic data of Inputs (I), Outputs (O), Pre-conditions (P) and Effects (E) of services and queries.

The non logic-based matching approaches use syntactic, structural and numeric mechanisms like syntactic similarity, term frequencies, numeric distance and structured graph matching. The main idea is to use implicit semantic rather than explicit one. DSD-matchmaker [6] and iMatcher1 [7] are examples of non logic-based matching approaches.

The hybrid matching approaches use a combination of logic and non-logic mechanisms. Ontology Web Language for Services matchmaker (OWLS-MX) [8], Web Service Modeling Ontology matchmaker (WSMO-MX) [5], Semantic Annotations for Web Services Description Language and XML Schema matchmaker (SAWSDL-MX) [9] are examples of hybrid matchmakers. The OWLS-MX matchmaker exploits both logic-based reasoning and content-based information retrieval techniques for OWL-S service profile I/O matching. The WSMO-MX matchmaker applies different matching filters to retrieve semantic web services. It computes logic-based and syntactic similarity-based matching degrees and returns a ranked set of services that are semantically relevant to a user request. The SAWSDL-MX matchmaker is inspired from OWLS-MX and WSMO-MX. It performs hybrid matching for SAWSDL operations based on both subsumption reasoning (logic-based matching) and text retrieval technique (IR-based matching). It combines the results to provide a matching result for service interfaces with multiple operations.

To our knowledge there are relatively few works focusing on description and matching strategy of multimedia web services. In a previous work [13], we have defined a general framework for our multimedia web service approach. The present work aims at extending the similar works to make further progress in multimedia web services matching.

3. Multimedia Ontologies

There is a strong need of annotating multimedia contents to enhance the agents' interpretation and reasoning for an efficient search on the web. It is well known that ontologies increases the precision of multimedia retrieval information systems. In last decade, significant research efforts have been made to build and implement multimedia ontologies. In [16] the authors compare well-known ontologies in the multimedia domain. The comparative study is done on 16 ontologies that are classified in four categories:

1. Multimedia objects in general.
2. Images and shapes as visual elements for representing images.
3. Visual objects in general.
4. Music ontologies.

We are interested by the ontologies of the first category because of their generic character. In order to represent multimedia knowledge of MaaSs, we adopt the media resource ontology [11]. This choice is justified by:

1. It is W3C recommendation that is developed by W3C Media Annotation Working Group.
2. It provides mappings with a variety of multimedia formats (Dublin Core, Learning Object Metadata (LOM) 2.1, Multimedia Content Description Interface (MPEG-7), Exchangeable Image File Format (EXIF), etc.,), which facilitates the interoperability.
3. It is well documented, which benefits the ontology understanding. In addition, this ontology covers all the multimedia aspects, it is the most general for describing multimedia objects.

4. MaaS Service Formalization

In this section, we formally define a MaaS service and a user query.

- *Definition 1*: MaaS serviceWe have extended the work of Vaculin *at el*. [17], which defines a service as a pair of inputs and outputs, by adding ontological concepts for annotating them. Formally, a MaaS web service is a 5-tuple:

    MaaS = (I, O, IC, OC, MC), where

- *I*: is the set of inputs, I={(?v,T) | ?v ∈ Var, T ∈ metierType}.
- O: is the set of outputs, $O=\{(?v,T) \mid ?v \in Var, T \in mediaType\}$, *Var* is the set of inputs and outputs names, *metierType* and *mediaType* are two types of XML Schema defined in our previous work [13],
- *IC*: is the set of concepts annotating the inputs of the service.
- *OC*: is the set of concepts annotating the outputs of the service.
- *MC:* is the set of multimedia concepts annotating the service.

- *Definition 2*: Query in the same way as MaaS services, we represent a query Q as:

    $Q = (I_q, O_q, \mathcal{O}, IC_q, OC_q, MC_q)$

Each term in Q has the same signification as inMaaS.

5. The Proposed Approach

The MaaS services (MaaSs) are specific data web services that access to multimedia data. In a previous

work [13], we have presented a full service approach to aggregate MaaSs for multimedia data retrieving.

5.1. MaaS Description Phase

Many languages and approaches have been developed with the goal to describe semantic web services. We distinguish two main classes of these approaches. Approaches of the first class are based on adding annotations, such as SAWSDL [10] and Universal Service-Semantics Description Language (USDL) [15]. Approaches of the second class are based on using of high-level ontology such as OWL-S [12] and WSMO [14], thus avoiding the problems of semantic heterogeneity that may occur. These last approaches use domain ontology to add semantic concepts in their description; they are a "closed approach": on the one hand, they manipulate a language ontology specification, e.g., Web Ontology Language (OWL) for OWL-S and Web Service Modeling Language (WSML) for WSMO. On the other hand, they specify very limited set of concepts that are not easily extensible. However, SAWSDL remains an independent approach to language semantic representation. This independence is ensured by the separation between the mechanisms of semantic annotation and representation of semantic description. Without such a mechanism, developers don't have enough flexibility to select their favourite semantic representation of languages or to reuse their own ontology to annotate services [2]. In addition, SAWSDL is close to WSDL, it does not require more effort for developers familiarized with WSDL. This is an important advantage compared to other approaches. For all these reasons, we choose SAWSDL language to annotate semantically MaaSs. In addition, we need to take into account the multimedia aspects (e.g., format, location, creation, etc.,). Consequently, the main idea of this work is to extend SAWSDL for enhancing expressiveness of multimedia service description. The use of model Reference attribute of SAWSDL to annotate MaaSs is not sufficient, this attribute allows to reference concepts describing a business domain of services. However, we need to reference separately the concepts defining the semantics of multimedia data of services. Then, we propose to add a new attribute: multimediaConcept. This attribute allows adding a new description level linked to multimedia aspects.

Our MaaS description approach includes two types of ontologies:

1. Domain Ontology containing concepts that covers a business domain (e.g., medical, tourism, etc.,).
2. MultiMedia Ontology containing concepts defining a set of annotation properties for describing multimedia content.

These properties are URIs of multimedia ontology objects. This means that the annotation of a concept in

our MaaS approach is the way to tie together this concept to a class that exists in ontology. The multimedia ontology used is Media Resource Ontology presented in the section above. The MaaS description was enriched by references to multimedia concepts such as: type of media resource, format, location, creation properties, etc.Therefore, we believe that it is necessary to be able to differentiate the semantic annotation of services capabilities and the semantic annotation of data provided by services. Differentiation of semantic annotations for MaaSs aspects can be used to enhance discovery and multimedia data searching in our full service approach.The MaaS description is structured in three layers:

1. Syntactic description based on WSDL standard.This layer corresponds to the terms I and Ofrom the MaaS definition.
2. Domain description represented by a set of annotations based on domain ontology. The SAWSDL *modelReference* attribute is used to add these annotations. We denote the domain description of any MaaSS_i by S_i.D.This layer corresponds to the terms IC and OC from the MaaS definition.
3. Mulimedia description represented by a set of annotations based on multimedia ontology. The SA4MaaS *multimediaConcept* attribute is used to add these annotations. We denote the multimedia description of any MaaSS_i by S_iM.This layer corresponds to the term MC from the MaaS definition.

The introduction of*multimediaConcept* attribute makes SAWSDL descriptions more expressive and significant for multimedia data retrieving. The search of a multimedia resource in our system is based either on annotations business domain, or on multimedia annotations or both.

5.2. MaaS Matching Phase

The purpose of this phase is to identify relevant MaaSs to meet user request, i.e., how to find MaaSs satisfying the query in the best way.To achieve this goal, we need to identify a new matching mechanism able to find a similarity between the domain description and the multimedia description of both MaaSs and user queries. We assume in this paper that the query is described in the same way as MaaSs.At the MaaS description phase, business concepts are specified through the attribute "modelReference" and multimedia concepts through the "multimediaConcept" attribute. The "modelReference" attribute specifies the concepts from the business domain ontology whereas the "multimediaConcept" attribute specifies the concepts from the multimedia ontology. Our proposed matching process is performed in two successive steps. The first step, "Domain matching", consists to

compare a domain description of MaaSs (*S.D*) with a domain description of query (*Q.D*). The second step, "Multimedia matching", consists to compare a multimedia description of MaaSs (*S.M*) with a multimedia description of query (*Q.M*). This step is performed if and only if the previous step has succeeded.

5.2.1. Domain Matching

This step focuses on the identification of relevant MaaSs to meet the user's request based on their domain description. This refinement is done by applying our matching mechanism (described below) between the query and the candidate MaaSs. In this work, we assume that MaaSs and the query are annotated using the same ontology. The domain concepts of input and output MaaSs and the query are extracted from their SAWSDL files. Our domain matching approach is based on an IO-matching, i.e., a matching process that considers only the inputs and outputs. The matching of these elements is summarized in a matching between the annotated concepts. We assume, for simplicity, that both input and output are annotated by a single concept. The similarity between two concepts (c_1, $c_2 \in C$) is evaluated by a matching degree. The different matching degrees used in our approach are:

- "*Exact*": if c_1 and c_2 are the same (or equivalent).
- "*Subsumed*": if c_1 is a sub-concept of c_2.
- "*Subsumed-by*": if c_2 is a sub-concept of c_1.
- "*Has-Relation*": if c_1 and c_2 are linked by a relation,
- "*Has-same-Hierarchy*": if c_1 and c_2 belong to the hierarchy of a same concept.
- "*Unknown*": if one of c_1 or c_2 is not specified.
- "*Fail*": If no relationship can be determined between c_1 and c_2.

- *Definition 3*: MaaS Similarity

  A MaaS similarity is a function *sim:*$C \times C \mapsto [0, 1]$, which calculates the similarity degree $sim(c_1, c_2) \in [0, 1]$ between two concepts c_1 ($c_1 \in C$) and c_2 ($c_2 \in C$).

Table 1. Similarity degrees.

Matching Degree	Exact	Subsumes	Subsumed-by	Has-Relation	Has-Same-Hierarchy	Unknown	Fail
Similarity Degree	1	4/5	4/5	3/5	2/5	1/5	0

We have extended the existing works of matching services by adding two degrees: "Has-Relation" and "Has-same-Hierarchy" that are specific to our approach. We associate with these matching degrees the numeric values between [0,1] (Table 1) representing similarity degrees, enabling the calculation of the similarity function SIM between a MaaS and a query.

This similarity degrees have not been arbitrarily chosen, they are based on the importance of the

semantic link between the elements to match. Strong semantic links are close to 1 and weak semantic links are close to 0. A similarity degree equal to 1 means that the matching is correct or equivalent, whereas a similarity degree equal to 0 means a failure of matching. We consider the two matching degrees "Subsumes" and "Subsumed-by" having the same similarity degree that equal to 0.8. Let's assume a set of MaaS {S_1, S_2... S_n} and a query Q. In order to match a MaaS with a user query, we have proposed the algorithm 1 (Domain matching) able to filter relevant MaaS by comparing their two domain descriptions.

```
1   Algorithm 1: Domain Matching
2   Inputs: Query Q, set of MaaS services S {s₁,s₂, ..., sₙ},
3   Threshold θ
4   Outputs: set of relevant MaaS services R {s₁,s₂, ..., sₘ}
5   /* the function Similarity returns the similarity degree
6   between two concepts. */
7   /* the function SIM returns the global similarity degree.
8   */
9   Begin
10      R ← ∅
11      for each service sᵢ in S do
12          SIM_IN ← Similarity(sᵢ.ic,Q.ic)
13          SIM_OUT ← Similarity(sᵢ.oc,Q.oc)
14          SIM ← (SIM_IN+SIM_OUT)/2
15          if ( SIM >= θ ) then
16              R ← R ∪ {sᵢ}
17          end if
18      end for
19      return R
20  End
```

In algorithm 1, the *SIM_IN* and *SIM_OUT* terms denote respectively the input similarity and the output similarity. *SIM_IN* is calculated between the input domain concept of the first service S_1 ($S_1.ic$) and the input domain concept of the query Q ($Q.ic$) (line 10). *SIM_OUT* is calculated between the output domain concept of the first service S_1 ($S_1.oc$) and the output domain concept of the query Q ($Q.oc$) (line 11). The global similarity function is calculated, *SIM = (SIM_IN + SIM_OUT) /2* (line 12). The same process is repeated for all remaining MaaSs *{S_2,S_3,..., S_n}*. At the end, only are kept the services that have a similarity measure *SIM* greater or equal than a threshold $\theta \in [0, 1]$ (θ is a numerical value chosen by the user).

5.2.2. Multimedia Matching

As we said before, a multimedia concept is used to describe metadata of multimedia content. In our approach, we represent this metadata as SPARQL query defined over multimedia ontology. The domain matching mechanism based on IO-matching model cannot be used to represent the metadata of the MaaSs because this model does not take into account the semantic relationships that may exist between the ontological concepts annotating the metadata part of a

MaaS. In this sense, we propose to annotate the metadata part of a MaaS with a declarative semantics represented by a SPARQL query. In a SPARQL query, the semantic relationships between the ontological concepts are described by *ObjectProperties*. Let us consider the example presented in Figure 1. It represents a SPARQL query that return the creator name (?x) and the location name (?y) of video files. In this query, the semantic relationships between concepts annotating metadata of MaaS are represented by the objectproperties: "hasCreator" and "hasLocation".

```
1   PREFIX rdf:"http://www.w3.org/1999/02/22-rdf-syntax-ns"
2   PREFIX ma:"http://www.w3.org/ns/ma-ont"
3   SELECT  ?x, ?y
4   WHERE{
5        ?M    rdf:type   ma:Video.
6        ?M    ma:hasCreator   ?C.
7        ?C    rdf:type   ma:Creator.
8        ?C    ma:hasName   ?x.
9        ?M    ma:hasLocation   ?L.
10       ?L    rdf:type   ma:Location.
11       ?L    ma:hasName   ?y.}
```

Figure 1. SPARQL query.

We use the principle of query containment [4] to enable the comparison between multimedia part of MaaS *(S.M)* and multimedia part of query *(Q.M)*. A *Q.M* is said to be contained in *S.M*, denoted by *Q.M⊆ S.M*, if and only if the answer to *Q* is a subset of the answer to *S* for any knowledge base. For this second step of matching, we have proposed the algorithm 2 (Multimedia matching) able to compare the multimedia description of MaaS with the multimedia description of user query. In the first part of algorithm (lines 9-18), we compare each class node C_Q in *Q* to each class node C_{Si} in the service $S_i.M$ and, if classes match, we continue the process. In the second part (lines 20-29), we check that all object properties in the query *Q* are covered by the metadata query of the service. The implementation of the functions *classNodeCovering()*, and *objectPropertyCovering()* is provided in the paper [1].

6. Experiments

We choose to validate our approach in the medical field. We are particularly interested in an illustrative example, on cancerous diseases data. We assume that a student in medicine wants to find videos about the different diagnostic methods of lung cancer disease. These videos are published by "cancercenter" company in "mpeg" format. Let's assume that this student has a set of MaaSs (Table 2) to respond to his request. For the description of these MaaSs, we have used two ontologies:

1. The cancer-onto ontology [13] used to add business domain concepts to the services.

2. The multimedia ontology described in section 3 and used to add multimedia concepts to the services.

The third and fifth columns of Table 2 represent the ontological concepts annotating respectively inputs and outputs of services. However, the last column represents the ontological concepts annotating the multimedia aspects of services. We annotate the user query by using the same principle. The result of this annotation is given in Table 3.

Table 3. Query annotation.

	Query Input	Query Output	Query Multimedia Concept
Q	Lung_cancer	LC_Diagnoses	Video, Publisher, Format

To validate our proposals, we have made an evaluation that shows the impact of multimedia matching on the system performance. For that, the experimental evaluation focuses on comparing our proposed approach with and without multimedia matching step. The evaluation is based on calculating the well-known measures recall, precision and F-Measure. Figures 2-a, 2-b, and 2-c give the graphical representation of Recall, Precision, F-Measure values for the two variants of our proposed approach.

```
1   Algorithm 2: Multimedia Matching

2   Inputs: Query Q, a MaaS services Sᵢ
3   Outputs: Boolean value isMatched (Match or no match)
4   Begin
5      classNode C_Q in Q;
6      objectProperty OP_Q in Q;
7      isMatched ← false;
8      C_Q.first();
9      do
10        classMatch ← false;
11        for each class node C_Si in  Sᵢ.M do
12           if (C_Q and C_Si have the same class type) then
13              if (classNodeCovering(C_Q,C_Si)) then
14                 classMatch← true;
15                 break;
16        end for
17        C_Q.next();
18     while (C_Q.hasnext() and classMatch)
19     OP_Q.first();
20     do
21        objPropertyMatch← false;
22        for each object property OP_Si in  Sᵢ.M do
23           if (OP_Q and OP_Si are the same) then
24              if (objectPropertyCovering(OP_Q,OP_Si)) then
25                 objPropertyMatch← true;
26                 break;
27        end for
28        OP_Q.next();
29     while (OP_Q.hasnext() and objPropertyMatch)
30     if (classMatch and objPropertyMatch) then
31           isMatched← true;
32     return isMatched;
33 End
```

Table 2. Medical MaaS services.

Services	Service name	Service Input	SIM_I	Service Output	SIM_O	Service Multimedia Concept
S_1	Treatmt_LC	Lung_ Cancer	1	LC_ Treatment	0	Image, Publisher(cancercenter), Format (jpeg)
S_2	Diagnoses_LC	Lung_ Cancer	1	LC_ Diagnoses	1	Video, Publisher(cancercenter), Format (mpeg)
S_3	Med-Imag _LC	Lung_ Cancer	1	Medical_ Imaging	0.8	Audio, Publisher(youtube), Format (wav)
S_4	Symptom_LC	Lung_ Cancer	1	LC_ Symptom	0	Video, Publisher(cancercenter), Format (mpeg)
S_5	Causes_LC	Lung_ Cancer	1	LC_ Causes	0	Image, Publisher(imaginis), Format (jpeg)
S_6	Diagnoses_PC	Prostate_ Cancer	0.4	LC_ Diagnoses	1	Video, Publisher(cancercenter), Format (mpeg)
S_7	Treatmt_Nsmall-cell	Non_Small_cell	0.8	LC_ Treatment	0	Image, Publisher(tripadvisor), Format (jpeg)
S_8	Diagnoses_Symptom	LC_ Symptom	0.6	LC_ Diagnoses	1	Video, Publisher(imaginis), Format (mp4)
S_9	Examint_Nsmall-cell	Non_Small_cell	0.8	Examination	0.8	Video, Publisher(cancercenter), Format (mpeg)

a)Recall.

b)Precision.

c)F-Measure.
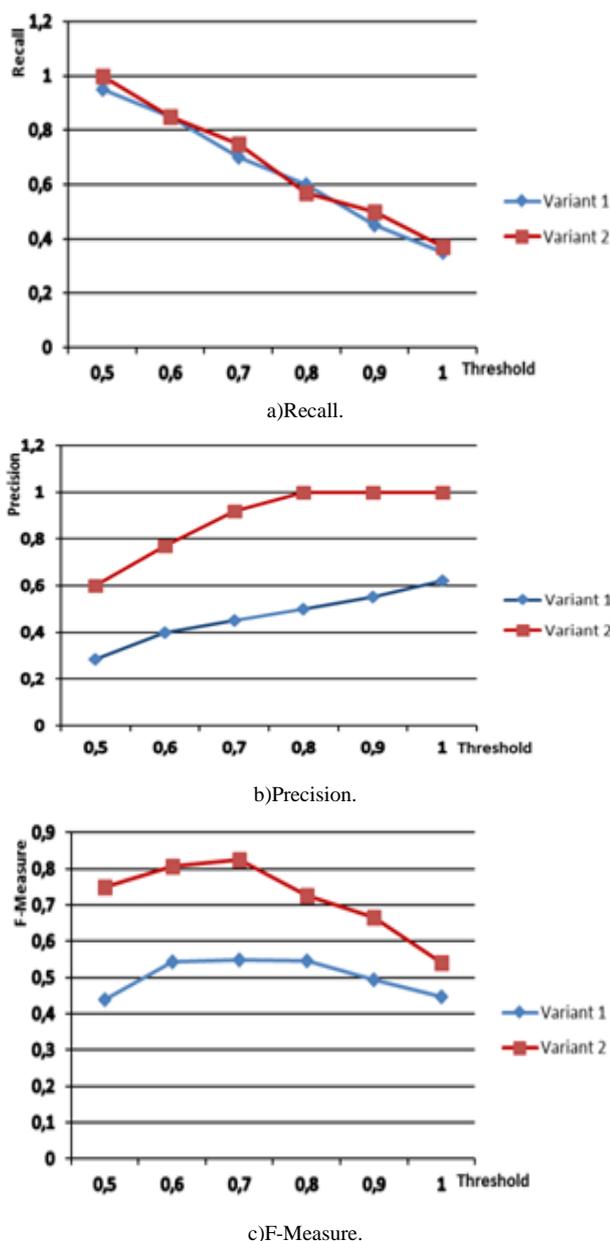
Figure 2. The system performance.

The first variant (variant 1 in graphics) represents the evaluation results of our approach without multimedia matching. However, the second variant (variant 2 in graphics) represents the evaluation results of our approach with multimedia matching. As seen in Figure 2-a, we notice that both variants have similar performance in terms of recall. On the other hand, we notice in Figure 2-b a large superiority of the second variant relative to the first in terms of precision. We remark an increasing in the precision of the second variant, which will be equal to 1. This situation is caused by the elimination of impertinent results retrieved by the system when using both the domain and multimedia matching. Equal consideration of recall and precision using the F-Measure yields the results given in Figure 2-c, which recapitulates the observations. In conclusion, the variant with multimedia matching offers better performance. The results show that the threshold range [0.7, 0.8] is a good compromise for the precision and recall. Our main challenge is then how to set the threshold when using a large test collection. We envisage in a near future to use some existing test collection in evaluating the proposed approach. However, up to our knowledge, there is no existing test collection adapted to our experiments. Hence, we need to prepare a big number of MaaSs for a consequent evaluation of our approach.

7. Conclusions and Future Work

In this paper, we presented our recent work and experiments on multimedia data retrieving. We have proposed an extension of SAWSDL for MaaSs. This extension uses two types of ontologies: Domain Ontology and multimedia ontology. The Domain Ontology references business domain concepts of the web service. The Multimedia Ontology references multimedia concepts defining a set of annotation properties for describing multimedia content. In addition, we have presented how this extension is used to address the problem of matching between MaaSs and user needs. To achieve this goal, we have proposed a new matching mechanism for MaaSs. An experiment is conducted to validate the new proposed approach. Results indicate that the use of both domain and multimedia matching considerably improve the performance of multimedia data retrieving systems.

This proposal is part of an ongoing work for implementing the MaaS framework through the development of a general architecture for MaaS

description, discovery and invocation. Future work will concern the specification of a language query facility together with mechanisms for querying and searching a multimedia content of MaaSs. We recall that MaaSs can return one or more image, video, audio or text files. Another challenge, arising from this diversity, consists to give the user homogeneous and coherent results. Otherwise said how to combine different types of MaaSs, such as an image MaaS with a video MaaS, to answer the user query. Accordingly, we believe that the proposed approach requires more reflection on both its theoretical and practical aspects. However, we need first to evaluate it in a large scale setting.

References

[1] Barhamgi M., Benslimane D., and Medjahed B., "A query Rewriting Approach for Web Service Composition," *IEEE Transactions on Services Computing*, vol. 3, no. 3, pp. 206-222, 2010.

[2] Chabeb Y., Tata S., and Ozanne A., "YASA-M: A Semantic Web Service Matchmaker," *in Proceedings IEEE 24th International Conference on Advanced Information Net-Working and Applications*, Perth, pp. 966-973, 2010.

[3] Hadjila F. and Chikh A., "Automated Retrieval of Semantic Web Services: A matching Based on Conceptual Indexation," *The International Arab Journal of Information Technology*, vol. 10, no. 1, pp. 61-66, 2013.

[4] Halevy A., "Answering Queries Using Views: A Survey," *The International Journal on Very Large Data Bases*, vol. 10, no. 4, pp. 270-294, 2001.

[5] Kaufer F. and Matthias K., "Wsmo-Mx: A logic Programming Based Hybrid Service Matchmaker," *in Proceedings of European Conference on Web Services*, Zurich, pp. 161-170, 2006.

[6] Klein M. and König-Ries B., "Coupled Signature and Specification Matching for Automatic Service Binding," *in Proceedings of Web Services*, Erfurt, pp. 183-197, 2004.

[7] Klusch M., "Semantic Web Service Coordination," *in Proceedings of CASCOM: Intelligent Service Coordination in the Semantic Web*, Birkhäuser Basel, pp. 59-104, 2008.

[8] Klusch M., Fries B., and Sycara K., "OWLS-MX: A hybrid Semantic Web service Matchmaker for OWL-S Services," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 2, pp. 121-133, 2009.

[9] Klusch M. and Kapahnke P., "Semantic Web Service Selection with SAWSDL-MX," *in Proceedings of the Second International Conference on Service Matchmaking and Resource Retrieval in the Semantic Web*, Karlsruhe, pp. 2-16, 2008.

[10] Lausen H. and Farrell J., "Semantic Annotations for WSDL and XML Schema," *W3C recommendation, W3C*, 2007available at:https://www.w3.org/TR/sawsdl/, Last Visited, 2017.

[11] Lee W., Bailer W., Bürger T., Champin P., Evain J., Malaisé V., Michel T., Sasaki F., Söderberg J., Stegmaier F., and Strassner J., "Ontology for Media Resources," *W3C recommendation*,2012, available at: http://www.w3.org/TR/mediaont-10/, Last Visited, 2017.

[12] Martin D., Burstein M., Hobbs J., Lassila O., McDermott D., and McIlraith S.,"OWL-S: Semantic Markup for Web Services," *W3C Member Submission*, vol. 22, no. 4, 2004.

[13] Midouni S., Amghar Y., and Chikh A., "A Full Service Approach for Multimedia Content Retrieval," *in Proceedings of International Conference on Model and Data Engineering*, Larnaca, pp. 125-137, 2014.

[14] Roman D., Keller U., Lausen H., De-Bruijn J., Lara R., Stollberg M., Polleres A., Feier C., Bussler C., and Fensel D., "Web Service Modeling Ontology," *Applied Ontology*, vol. 1, no. 1, pp. 77-106, 2005.

[15] Simon L., Mallya A., Bansal A., Gupta G., and Hite T., "A Universal Service Description Language," *in Proceedings IEEE 20th International Conference on Web Services*, Orlando, pp. 823-824, 2005.

[16] Suárez-Figueroa M., Atemezing G., and Corcho O., "The Landscape of Multimedia Ontologies in the Last Decade," *Multimedia Tools and Applications*, vol. 62, no. 2, pp. 377-399, 2013.

[17] Vaculín R., Chen H., Neruda R., and Sycara K., "Modeling and Discovery of Data Providing Services," *in Proceedings of IEEE International Conference on Web Services*, Beijing, pp. 54-61, 2008.

Sid Midouni is PhD candidate at INSA-Lyon (France) and UABT University of Tlemcen (Algeria). He received a MSc from INSA-Lyon in Computer Science in 2005 and Magister degree in computer science from UABT University of Tlemcen in 2008. His research interests include information retrieval, web services, semantic web and multimedia data.

Youssef Amghar received his PhD degree in 1989. He begins his career as assistant Professor in computer science at INSA-Lyon from 1990 to 2003. He becomes full Professor starting from 2003 and leads a research team in many fields of computer sciences (databases, information systems, services, and multimedia data). His is author of more than 100 publications both in journals and conferences.

Azeddine Chikh holds a PhD in Information Systems from National Institute of Computer Sciences (NICS) in Algeria and Paul Sabatier University in France in 2004, a Masters in Information Systems from NICS in Algeria 1994, a Masters in e-learning from Switzerland in 2007, and a graduate certificate in systems engineering from USA, in 2003. His research interests include learning design; communities of practice; social networks; semantic web; and software engineering.