Stefan Schulte, Ulrich Lampe, Matthias Klusch, Ralf Steinmetz:
COV4SWS.KOM: Information Quality-aware Matchmaking for Semantic Services.
In: Elena Simperl, Philipp Cimiano, Axel Polleres, Oscar Corcho, Valentina Presutti: Proceedings of the 9th Extended Semantic Web Conference 2012 (ESWC 2012), p. 499-513, Springer-Verlag, May 2012.
ISBN 978-3-642-30283-1.

COV4SWS.KOM: Information Quality-aware Matchmaking for Semantic Services

Stefan Schulte[1], Ulrich Lampe[2], Matthias Klusch[3], and Ralf Steinmetz[2]

[1] Distributed Systems Group, Vienna University of Technology, Austria
s.schulte@infosys.tuwien.ac.at
[2] Multimedia Communications Lab, Technische Universität Darmstadt, Germany
[3] German Research Center for Artificial Intelligence, Saarbrücken, Germany

Abstract. The discovery of functionally matching services – often referred to as matchmaking – is one of the essential requirements for realizing the vision of the Internet of Services. In practice, however, the process is complicated by the varying quality of syntactic and semantic descriptions of service components. In this work, we propose COV4SWS.KOM, a semantic matchmaker that addresses this challenge through the automatic adaptation to the description quality on different levels of the service structure. Our approach performs very good with respect to common Information Retrieval metrics, achieving top placements in the renowned Semantic Service Selection Contest, and thus marks an important contribution to the discovery of services in a realistic application context.

1 Introduction

From the very beginning of semantic Web service (SWS) research, service discovery and matchmaking have attracted large interest in the research community [9, 13, 18]. The underlying techniques to measure the similarity between a service request and service offers have been continuously improved, but matchmakers still rely on a particular information quality regarding the syntactic and semantic information given in a service description. There are several reasons why the quality of syntactic and semantic service descriptions differs between service domains. While in one domain, a well-accepted ontology describing the particular (industrial) domain could be available, such an ontology might be missing for other domains. Furthermore, it could be the case that the usage of a certain domain ontology in a specific industry is compulsory due to legal constraints, as it is the case in the energy domain. In an upcoming *Internet of Services*, it is even possible that there will be premium service marketplaces for certain domains, which will only publish a service advertisement if certain quality standards regarding the service description are met. All things considered, the quality of service descriptions will differ from service domain to service domain.

In this paper, we present our work on information quality-aware service matchmaking. We propose an adaptation mechanism for matchmaking, which is based on the usability and impact (with regard to service discovery) of syntactic descriptions and semantic annotations on different levels of the service description structure. The actual measurement of semantic-based similarities is based on metrics from the field of information theory. Furthermore, a linguistic-based fallback strategy is applied. Our approach is implemented in COV4SWS.KOM, a service matchmaker for WSDL 2.0 and SAWSDL. Unless explicitly defined otherwise, it is possible to transfer all concepts and results from this paper to WSDL 1.1 based on corresponding mappings respectively adaptations of the algorithms. In fact, COV4SWS.KOM also operates on other SWS description standards, most importantly hRESTS in conjunction with MicroWSMO for RESTful services. However, due the existence of a de facto standard test collection for the purposes of evaluation, our focus lies on WSDL in combination with SAWSDL in the work at hand. COV4SWS.KOM extends our former work on semantic matchmaking [14, 22] by providing a novel approach to adapt matching results based on the usability of service descriptions in a certain domain.

The remainder of this paper is structured as follows: In Section 2, we provide a brief presentation of SAWSDL. Our general considerations as well as the matchmaking approach are presented in Section 3. We evaluate different configurations of COV4SWS.KOM and compare the results with other matchmaking approaches for SAWSDL (Section 4). As will be presented in the evaluation, COV4SWS.KOM is capable of competing with state-of-the-art matchmakers in terms of Information Retrieval (IR) metrics like precision and recall while offering an adaptation mechanism for different degrees of description quality. Eventually, we comment on the related work (Section 5) and conclude this paper (Section 6).

2 Service Descriptions using WSDL 2.0 and SAWSDL

To keep this paper self-contained, we will give a short discussion of WSDL 2.0 in the following. We refer to the WSDL 2.0 specification for further details [3].

The abstract part of a WSDL document advertises *what* a service does while the concrete part defines *how* a service can be consumed and *where* it is located. In service discovery, the description of what a service does is of primary interest. Hence, in the following, the abstract part of a WSDL-based service description – interfaces, operations, and message parameters – will be utilized. These service components constitute the *service abstraction levels* of WSDL 2.0: Functionalities, i.e., interactions between a client and a service, are described by abstract operations. A set of operations defines a service interface. For each operation, a sequence of messages a service is able to send or receive may be defined. In WSDL 2.0, messages are defined using (XSD) parameter types [3].

SAWSDL imposes neither restrictions on what a semantic annotation eventually means nor the type of semantic concept that is addressed. However, the SAWSDL specification states that on interface level, a *modelReference* might be a categorization, while on operation level, a modelReference might specify a high level description of the operation – both apply to functional semantics as defined by Gomadam et al. [7]. On message respectively parameter level, modelReferences most likely define data semantics [6]. This meaning of semantic annotations is compliant with the classification made for *WSMO-Lite* [23] and will be the foundation for the matchmaking approach presented in this paper.

Likewise, 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. This is an advantage in so far as it allows for a maximum of flexibility in annotating functional service descriptions. Yet, this fact poses a problem if the concepts need to be automatically processed and interpreted in some form. In the context of our work, we will assume the semantic concepts to be formally defined in an OWL DL ontology. As a second constraint, we only consider the first URI from a modelReference, all other URIs are not regarded. This constraint is primarily made for practical reasons, as there is no agreement what another modelReference actually addresses: It could be a reference to a semantic concept from another domain ontology or address preconditions and effects, as it is done for operations in WSMO-Lite [23].

3 Information Quality-based Matchmaking

As our matchmaking approach aims at the information quality-aware adaptation of the service discovery process, there are a number of challenges to be met: First, it is necessary to provide the means to compute both syntax- and semantic-based similarity values for different service components, i.e., interfaces, operations, inputs, and outputs. Second, the according similarity values need to be easily combinable in order to derive an overall value for a service request and the service offers that come into consideration. We will investigate these aspects in detail in Section 3.2. Third, the information quality needs to be assessed. Therefore, it is necessary to define what the meaning of information quality actually is:

Information quality of service descriptions. According to [24], data or information quality is a subjective value that needs to be assessed with regard to the task the information will be applied to. Hence, in service matchmaking, information quality depends on the positive impact a service description will have on the outcome of the actual matchmaking process: If a certain information will have a relatively large positive impact on the identification of relevant services, its quality is assumed to be relatively high. Therefore, it is necessary to measure this impact and consequently adapt a matchmaker in order to give information with a higher quality a larger influence on the overall matchmaking results. With respect to service descriptions, information quality could also be interpreted as the degree to which the description is correct regarding the service it purports to describe. However, this would mean to interpret information quality with regard to service description *correctness*, not service discovery.

Information quality for service selection. As mentioned above, it is reasonable to assume different qualities of information in different service domains. While in one domain, there might be no standard for semantic information at all, there are other domains which provide some semantics (e.g., compulsory data semantics as in the already mentioned energy domain) and maybe – in the upcoming Internet of Services – even domains which demand to semantically describe the complete structure of a service description. Hence, it is reasonable to assume different qualities of semantic and syntactic information on the single abstraction levels of the service description structure. To let a matchmaker learn how to best select relevant services based on the respective degrees of information quality of these descriptions, a matchmaker's adaptation mechanism should be explicitly based on the positive impact the similarity values from different service abstraction levels will have on the retrieval results. This implies that information from all service abstraction levels should be computed and consequently combined using a weighting that indicates the impact of each level. We allow this by providing the means to automatically learn an optimal weighting based on an offline learner and providing similarity metrics for semantic and syntactic descriptions. The following paragraphs give an overview on COV4SWS.KOM, which will be further discussed in Sections 3.1 to 3.3:

Determination of similarities. For a service request and given service offers, COV4SWS.KOM returns a result set arranged in descending order regarding the computed similarity between request and offers. Information from all levels of the description structure is taken into account to calculate the overall similarity. For this, COV4SWS.KOM determines either the semantical or the syntactical similarity for different abstraction levels of a service description and aggregates the single values (cp. Section 3.2). As operations provide the essential functionality a service requester is looking for, COV4SWS.KOM makes use of an operations-focused, weighted aggregation of similarity values (cp. Section 3.1).

Aggregation of similarities. To combine the similarity results from the different service abstraction levels, COV4SWS.KOM performs a linear regression analysis using an Ordinary Least Squares (OLS) estimator (cp. Section 3.3) [25]. We assume that the weighted linear combination of similarities on individual levels predicts the aggregated similarity of two operations, and thus, ultimately, two services. The estimator learns the optimal weightings of abstraction levels during an offline training phase. For this, a selection of service requests and service offers, along with their respective mutual relevance rating, is available for training. A subset of a test collection satisfies this condition (cp. Section 4.1).

3.1 Operations-focused Matching

Generally, for each service request, the most relevant service offers should be identified. Following an operations-focused matching approach, the overall similarity between a service request and a service offer relates to the degree to which their respective operations match. This actually means that for each operation requested, the best matching operation in a service offer should be identified.

This leads to our overall matchmaking process as depicted in Figure 1. For each pair of operations in service request and offer, their respective input (sim_{in}), output (sim_{out}), native operation (sim_{op}), and interface (sim_{iface}) level similarity is computed using the similarity metrics presented in Section 3.2. This means that semantic or syntactic similarity is measured at every single service abstraction level based on the metrics described below. These individual similarities are then combined using specified weights, w_{in}, w_{out}, w_{op}, and w_{iface},

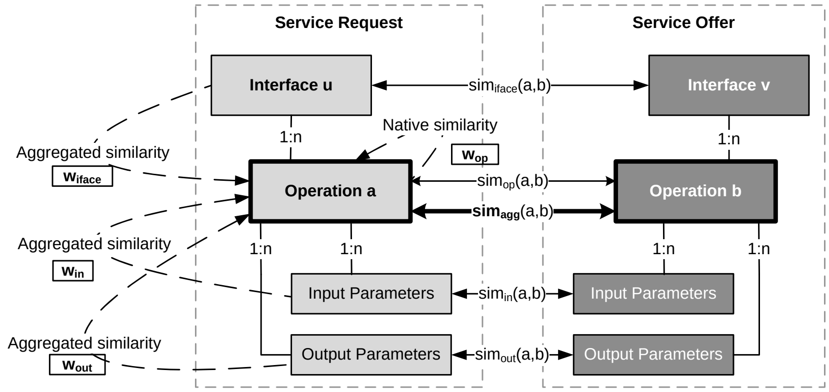

Fig. 1. Matchmaking Process in COV4SWS.KOM

resulting in an aggregated similarity value sim_{agg} for each pair of operations. Formally, for a pair of operations a and b, we define:

$$w_{iface} + w_{op} + w_{in} + w_{out} = 1 \tag{1}$$

$$\begin{aligned} sim_{agg}(a,b) = \quad & sim_{iface}(a,b) * w_{iface} + sim_{op}(a,b) * w_{op} \\ & + sim_{in}(a,b) * w_{in} + sim_{out}(a,b) * w_{out} \end{aligned} \tag{2}$$

Once similarities between all pairs of operations in a service request and service offer have been computed, the overall service similarity sim_{serv} is derived by finding an optimal matching of operations: The final matching for a pair of services is conducted between their respective union set of operations, disregarding how the operations are organized into interfaces. Formally, let I and J be the sets of operations in a service request R and offer O, respectively. Let x_{ij} be a binary variable, indicating whether $i \in I$ has been matched with $j \in J$. Then,

$$sim_{serv}(R,O) = \frac{1}{|I|} * \sum_{i \in I, j \in J} x_{ij} * sim_{agg}(i,j) \tag{3}$$

The matching of sets of components (specifically, inputs, outputs, and operations) is based on bipartite graphs. It perceives the sets of components of a service request and offer as two partitions of nodes in a graph. Each node in the first partition is connected with each node in the second partition through a weighted edge. The edge weights correspond to the respective similarity between two components. Using the well-established *Hungarian* (or *Kuhn-Munkres*) algorithm, the bipartite graph matching algorithm computes a 1-on-1 assignment of components. Each component of the service request is matched with one component of the service offer while maximizing the overall edge weight. If a component i has been matched with a component j, $x_{ij} = 1$, else $x_{ij} = 0$. To handle differing cardinalities of the sets, an extension of the Hungarian algorithm is applied [4].

Subsequent to the matching process, the weights of all matched edges are summed up and divided by the cardinality of the original sets. This yields the similarity for two sets of components. If the cardinality of the two sets differs, the following strategy is used: Generally, the cardinality of the set associated with the service request is decisive: If an offer lacks requested operations or outputs, its overall similarity decreases. For inputs, the cardinality of the set associated with the service offer is decisive: If an offer requires more inputs than the request provides, its overall similarity decreases. Such procedure does not exclude any services due to a mismatch in the number of parameters or operations. Instead, these offers are implicitly punished by a reduction in overall similarity. The approach is based on the notion that such service offers may still be able to provide a part of the initially requested functionality or outputs, or may be invoked by providing additional inputs.

3.2 Assignment of Similarities

In almost every case, the foundation for semantic-based matchmakers is subsumption reasoning, i.e., the determination of subconcept and superconcept relationships between semantic concepts [1]. Subsumption-based matchmaking suffers from a number of drawbacks. For one, it may reward the annotation with overly generic concepts and thus may lead to suboptimal matchmaking results [2]. This makes it necessary to use further aspects to penalize overly generic annotations. Second, subsumption-based Degrees of Match (DoMs) are quite coarse-grained and do not incorporate additional information available from the ontology structure, such as the distance between two concepts or the degree of increasing specialization between levels. Third, the combination with usually numerical similarity values from IR is generally not easy to achieve but nevertheless necessary in the work at hand as described above. Last, subsumption-based DoMs rely on a ranking, which can to some degree be quite arbitrary [22].

Hence, we make use of a different approach to compute the similarity between two semantic concepts in an ontology, the so-called *semantic relatedness*. The assignment of semantic relatedness of concepts in an ontology or taxonomy is a well-known problem from computational linguistics and artificial intelligence. In contrast to the logic-based subsumption matching usually applied, non-logic-based semantic relatedness possesses a certain degree of uncertainty, as it is the case with IR-based similarity measures: semantically related objects might still not be similar and may lead to both false positives and false negatives [5]. Nevertheless, such approaches provide a multitude of well-explored methodologies and similarity measures. Thus, methods from the field of semantic relatedness might provide a significant contribution to SWS matchmaking. In fact, it has been shown elsewhere that hybrid semantic service matchmaking which combines means of logic-based and non-logic-based semantic matching can outperform each of both significantly [10].

It is quite common in the area of service matchmaking to make use of a graph representation of an ontology, where the graph nodes represent the semantic concepts and the links (edges) between the nodes represent relationships between the concepts. The most intuitive way to compute semantic relatedness between nodes in a graph would be the measurement of the shortest distance (*path length*) between the graph nodes [5]. In the following, we refer to this measure as sim_{PL}. Furthermore, we make use of the metrics by Resnik [20] and Lin [15]:

$$sim_{Resnik}(A, B) = -\log p(anc(A, B)) \tag{4}$$

$$sim_{Lin}(A, B) = \frac{2 * \log p(anc(A, B))}{\log p(A) + \log p(B)} \tag{5}$$

For these metrics, all concepts in an ontology are augmented with values which indicate the *probability* that an instance of a concept is encountered (e.g., in a service description). The actual similarity of two concepts A and B is based on the probability p assigned to their most informative ancestor $anc(A, B)$. Probabilities are monotonically non-decreasing if moving up the taxonomy; if an ontology possesses a unique top node, its probability is 1. This can be traced back to the fact that classes inherit the probability values of their subclasses. After the probability has been determined, it is possible to derive the *information content* of $p(anc(A, B))$ which is defined as its negative log likelihood [5].

If there are no semantic concepts associated with service components or their processing fails, it might still be possible to measure the syntactic similarity of these components. E.g., it could be the case that on the message parameter level, types are semantically defined, while on operation and interface levels, only the syntax-based names of the components are available. Hence, we include a basic fallback strategy into our matchmaking approach. More precisely, the similarity between associated concept (and alternatively, component) names for a given pair of components is computed using the *WordNet* ontology [16]. Analogue to the semantic-based similarity measures, the similarity is a numerical value. Before the actual similarity can be computed, all names are tokenized. Tokens that do not correspond to a word in the WordNet ontology are additionally scanned for meaningful substrings in a recursive manner. Each set of words constitutes a partition for a bipartite graph. The edge weight corresponds to the inverse distance of a pair of words in WordNet. Consecutively, bipartite graph matching is employed, with the average edge weights in the matching yielding the similarity of the two names and thus, of two service components.

To improve the performance of matchmaking in terms of query response time and scalability, we utilize caching, namely semantic similarity, WordNet distance, and word splitting caches. Caches may be filled both at registration and query time. In the first and generally applied case, each new service offer is matched against all service offers in the repository, thus maximizing the cache population and, subsequently, the potential cache hit rate. In the latter case, only the results from matching the service queries against all service offers are stored.

3.3 OLS-based Automatic Weight Adaption

The central question regarding information quality-aware service discovery is to which degree different abstraction levels of a service description need to be regarded in the matchmaking process. As presented in Section 3.1, we allow the weighting of similarity values for interfaces, operations, and input and output parameters. The manual determination of such weightings is to some degree arbitrary. Furthermore, a particular weighting might be suitable for one service domain, as it reflects the information quality on the different levels of the service description structure correctly, but completely wrong for another service domain.

To account for this, COV4SWS.KOM applies an OLS estimator [25] for the determination of optimal level weights. The process is based on the notion that a dependent variable $y^{a/b}$, corresponding to the similarity of two operations a and b according to a numerical scale, can be derived through the linear combination of a set of independent variables $x_L^{a/b}$, corresponding to the individual similarity on a certain service abstraction level L when matching a and b.

In the training phase, COV4SWS.KOM matches all pairs of operations in all service requests and offers. For each pair a and b, the computed similarity on each level L yields a new entry $x_L^{a/b}$ for the *design matrix* X. Furthermore, COV4SWS.KOM retrieves the predefined similarity of operations a and b, which yields a new entry $y^{a/b}$ in the *vector of predictors* y. An example with realistic values is provided in Eq. 6 (a and b are operations in the service request; c, d and e are operations in the service offer; y is based on 4-point graded relevance).

$$(X|y) = \left(\begin{array}{cccc|c} x_{iface}^{a/c} & x_{op}^{a/c} & x_{in}^{a/c} & x_{out}^{a/c} & y^{a/c} \\ x_{iface}^{a/d} & x_{op}^{a/d} & x_{in}^{a/d} & x_{out}^{a/d} & y^{a/d} \\ x_{iface}^{a/e} & x_{op}^{a/e} & x_{in}^{a/e} & x_{out}^{a/e} & y^{a/e} \\ x_{iface}^{b/c} & x_{op}^{b/c} & x_{in}^{b/c} & x_{out}^{b/c} & y^{b/c} \\ x_{iface}^{b/d} & x_{op}^{b/d} & x_{in}^{b/d} & x_{out}^{b/d} & y^{b/d} \\ x_{iface}^{b/e} & x_{op}^{b/e} & x_{in}^{b/e} & x_{out}^{b/e} & y^{b/e} \end{array}\right) = \left(\begin{array}{cccc|c} 0.73 & 0.63 & 0.81 & 0.79 & 0.67 \\ 0.54 & 0.55 & 0.35 & 0.47 & 0.33 \\ 0.95 & 0.85 & 0.67 & 0.63 & 1.00 \\ 0.33 & 0.11 & 0.26 & 0.29 & 0.33 \\ 0.56 & 0.23 & 0.61 & 0.45 & 0.33 \\ 0.11 & 0.11 & 0.33 & 0.35 & 0.33 \end{array}\right) \tag{6}$$

Given the design matrix and vector of predictors, the standard OLS estimator can be applied [25]. It yields the initial estimate of level weights, namely the vector $\hat{\beta}$ (Eq. 7). In order to derive the final level weights, we further process the vector. First, negative level weights, which can potentially result from the OLS estimator, are set to 0, resulting in $\tilde{\beta}$ (Eq. 8). This ensures that increasing similarities on the individual levels do not have a negative impact on the aggregated similarity as it would be contradictory to common sense if higher similarity on one level resulted in diminished overall similarity. Second, the entries are normalized such that their sum matches the maximum relevance, resulting in the final vector w (Eq. 9). This ensures that a pair of operations with perfect similarity on all matching levels is precisely assigned the actual maximum relevance.

$$\begin{aligned} \hat{\beta} &= (X'X)^{-1}X'y = \left(\hat{\beta}_{iface}, \hat{\beta}_{op}, \hat{\beta}_{in}, \hat{\beta}_{out}\right) \\ &= \left(-0.063, 0.401, 0.506, 0.197\right) \end{aligned} \tag{7}$$

$$\begin{aligned} \tilde{\beta} &= \left(\min(0, \hat{\beta}_{iface}), \min(0, \hat{\beta}_{op}), \min(0, \hat{\beta}_{in}), \min(0, \hat{\beta}_{out})\right) \\ &= \left(\tilde{\beta}_{iface}, \tilde{\beta}_{op}, \tilde{\beta}_{in}, \tilde{\beta}_{out}\right) = \left(0, 0.401, 0.506, 0.197\right) \end{aligned} \tag{8}$$

$$\begin{aligned} w &= \left(\tilde{\beta}_{iface}/s,\, \tilde{\beta}_{op}/s,\, \tilde{\beta}_{in}/s,\, \tilde{\beta}_{out}/s\right) \\ &= \left(w_{iface},\, w_{op},\, w_{in},\, w_{out}\right) = \left(0,\, 0.363,\, 0.458,\, 0.178\right) \\ s &= \tilde{\beta}_{iface} + \tilde{\beta}_{op} + \tilde{\beta}_{in} + \tilde{\beta}_{out} \end{aligned} \tag{9}$$

To summarize, the essential idea of the OLS estimator is to approximate each level's impact on the overall service matching result, based on the computed similarities for the different matching levels. Thus, the matchmaker can dynamically account for missing or non-discriminatory semantic annotations or syntactic descriptions on certain matching levels. The application of OLS does not inflict the runtime performance of COV4SWS.KOM, because new level weights are only learned offline once new services are added to a repository. In our evaluation involving 42 requests and 1080 offers (cp. Section 4), the learning process can be conducted in the magnitude order of ten milliseconds. In general, with n_r denoting the number of requests and n_o denoting the number of offers, the worst-case computational complexity of OLS corresponds to $\mathcal{O}(n_r * n_o)$.

4 Experimental Evaluation

4.1 Evaluation Setup

The matchmaking approach presented in Section 3 has been implemented in *COV4SWS.KOM* using Pellet 2.0[4] as reasoner and JWNL 1.4[5] as interface to WordNet. COV4SWS.KOM is available as part of the XAM4SWS project[6].

As test data collection, SAWSDL-TC3[7] has been adopted. SAWSDL-TC3 consists of 1080 semantically annotated WSDL 1.1-based Web services, which cover differing domains. The set contains 42 queries. A service request is defined as a service that would perfectly match the request, i.e., requests and offers are both encoded using the same formalism. Furthermore, a binary and graded relevance set for each query is provided which can be used in order to compute IR metrics. As SAWSDL-TC3 is WSDL 1.1-based, it was necessary to convert the test collection to WSDL 2.0, which is the designated service format in the work at hand. For this, a XSLT stylesheet was created[8], based on a prototypical conversion tool by the W3C[9]. Because the conversion process does not add or remove any semantic or syntactic information, the resulting test collection can serve as a basis for comparison with WSDL 1.1 matchmakers.

In SAWSDL-TC, semantic annotations exist solely at message parameter level. As discussed in Section 3, COV4SWS.KOM incorporates information from the interface, operation, and message parameter levels of SAWSDL. This means

[4] `http://clarkparsia.com/pellet/`
[5] `http://jwordnet.sourceforge.net/`
[6] `http://projects.semwebcentral.org/projects/xam4sws`
[7] `http://www.semwebcentral.org/projects/sawsdl-tc`
[8] `http://www.kom.tu-darmstadt.de/~schulte/wsdl11to20.xsl`
[9] `http://www.w3.org/2006/02/wsdl11to20.xsl`

that the full potential of COV4SWS.KOM will only be revealed if the annotations address all service abstraction levels. However, SAWSDL-TC is a standard test collection for SWS matchmaking and needs to be employed to accomplish comparability with the results of other approaches. SAWSDL-TC is also used in the *International Semantic Service Selection Contest – Performance Evaluation of Semantic Service Matchmakers* (S3 Contest) [13], which serves as an annual contest to compare and discuss matchmakers for different service formalisms. Nevertheless, we assess our evaluation to be preliminary. We used SME2[10] to compare our results with other state-of-the-art matchmaking algorithms.

We performed evaluation runs using different configurations of our matchmaker; due to space constraints, we will only present the most important evaluation runs in the following. The interested reader can download the XAM4SWS matchmaker project to conduct evaluation runs using different configurations of COV4SWS.KOM. The applied configurations are depicted in Table 1; they make use of different weightings of service abstraction levels on matchmaking results and either apply sim_{Resnik}, sim_{Lin}, or sim_{PL}, as presented in Section 3.2.

For the OLS-based computation of weightings, the actual weights are identified using k-fold cross-validation [17]. In cross-validation, k–1 partitions of a test data collection are applied for training purposes (i.e., the determination of weights) while the remaining partition is applied for testing purposes (i.e., matchmaking). This is repeated k times in order to apply every partition in testing; validation results are averaged over all rounds of training and testing. In the example at hand, k=42 since every query and corresponding relevance set from SAWSDL-TC serves as a partition from the service set. The necessary probability values for sim_{Resnik} and sim_{Lin} have been calculated based on SAWSDL-TC, i.e., we counted the appearances of semantic concepts in the service collection and derived the probabilities from this observation.

4.2 Applied Metrics

In accordance with the procedure in the S3 Contest, we evaluated the IR metrics automatically computed by SME2, namely *Average Precision* (AP'), *Q-Measure* (Q'), and *normalized Discounted Cumulative Gain* (nDCG') for each configuration of COV4SWS.KOM [13, 21]. While AP' is based on binary relevance, Q' and nDCG' aim at graded relevance. As the apostrophes indicate, all numbers are adapted for incomplete relevance sets, i.e., there may exist relevant services which are *not* part of the relevance sets [21]. We deliberately refrain from the inclusion of *Average Query Response Time* (AQRT) in the evaluation results. In our opinion, the characteristics of the computer that is used for evaluation and the utilization of caches renders absolute AQRT figures largely incomparable. However, we refer the interested reader to the summary slides of the 2010 S3 Contest [13]. The summary provides a comparison of multiple matchmakers regarding the criterion of runtime performance and ranks COV4SWS.KOM – along with our other matchmaker, LOG4SWS.KOM [22] (also included in the

[10] `http://projects.semwebcentral.org/projects/sme2/`

Table 1. Summary of Evaluation Results for COV4SWS.KOM

COV4SWS.KOM Version No.	Weighting of Levels	Similarity Metric Applied	AP'	Q'	nDCG'
1a	0.0, 0.0, 0.5, 0.5	sim_{Lin}	0.710	0.725	0.787
1b	0.0, 0.0, 0.5, 0.5	sim_{Resnik}	0.734	0.708	0.760
1c	0.0, 0.0, 0.5, 0.5	sim_{PL}	0.755	0.770	0.828
2a	0.1, 0.1, 0.4, 0.4	sim_{Lin}	0.784	0.806	0.873
2b	0.1, 0.1, 0.4, 0.4	sim_{Resnik}	0.796	0.791	0.851
2c	0.1, 0.1, 0.4, 0.4	sim_{PL}	0.806	0.825	0.878
3a	0.25, 0.25, 0.25, 0.25	sim_{Lin}	0.796	0.812	0.867
3b	0.25, 0.25, 0.25, 0.25	sim_{Resnik}	0.808	0.808	0.869
3c	0.25, 0.25, 0.25, 0.25	sim_{PL}	0.806	0.825	0.881
4a	OLS-based	sim_{Lin}	0.802	0.813	0.877
4b	**OLS-based**	$\boldsymbol{sim_{Resnik}}$	**0.823**	**0.825**	**0.884**
4c	OLS-based	sim_{PL}	0.801	0.812	0.877

XAM4SWS project) – as the fastest contestant in the SAWSDL track. This can be traced back to the caching mechanisms applied (cp. Section 3.2).

4.3 Results and Discussion

Table 1 shows the evaluation results for the above mentioned configurations of COV4SWS.KOM. The evaluation led to somewhat heterogeneous results. Nevertheless, it is possible to derive some very important conclusions. If we compare the single similarity metrics, sim_{Lin} provides mostly better results than sim_{Resnik} for Versions 1 and 2; results for Version 3 are similar with a difference smaller than 0.01. For these three versions, sim_{PL} leads to the overall best results. Regarding the different versions, the signature-based Version 1 exhibits the worst results with respect to the metrics depicted in Table 1. The integration of similarity values from all service abstraction levels in Versions 2 to 4 clearly leads to an improvement of matchmaking results. Version 3 features better results than Version 2, i.e., the higher the weights for the interface and operation levels, the better the evaluation results. This shows that syntactically described service components make a very important contribution to the overall discovery results. Apart from sim_{PL}, the OLS-based Version 4 exhibits the best evaluation results, including the best overall results in Version 4b. Regarding sim_{PL}, the differences between Versions 2c, 3c, and 4c, are not significant.

Figure 2 shows the sim_{Resnik}-based versions (apart from Version 1b) performing quite similar over all recall levels. However, Version 4b provides better precision for recall levels 0.2-0.55, thus explaining the better results for this version. Figure 2 also shows very nicely that the inclusion of information from all service abstraction levels leads to improvements of results on all recall levels.

We have also compared COV4SWS.KOM with the most relevant contestants from the S3 Contest 2010, i.e., [8, 11–13, 19, 22]. The evaluation results of these matchmakers are depicted in Table 2. Apart from the AP', our own matchmakers LOG4SWS.KOM and COV4SWS.KOM provide the best results of all current

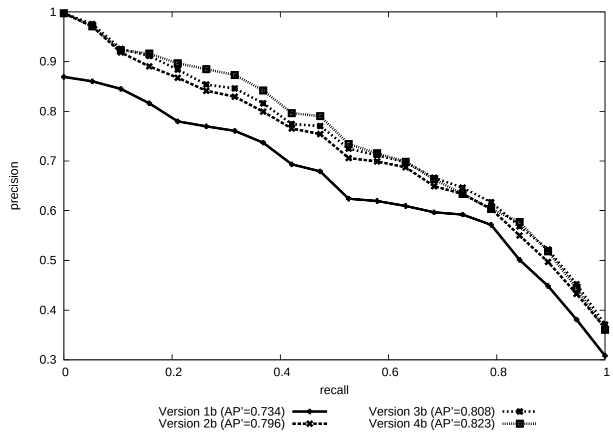

Fig. 2. Recall-Precision-Curves for sim_{Resnik}-based Versions

SAWSDL matchmakers. Based on a Friedman test (with a level of significance of *p=0.05*) [12], the differences for Q' and nDCG' for LOG4SWS.KOM and COV4SWS.KOM are not significant. Hence, the performance of LOG4SWS.KOM and COV4SWS.KOM regarding graded relevance sets (which are currently the state-of-the-art to assess retrieval algorithms in the IR community [21]), is estimated to be equal. So, we were able to show that the automatic adaptation provided by COV4SWS.KOM does provide very competitive matchmaking results: The matchmaker presented in this paper is one of the two current top matchmakers with regard to the evaluated IR metrics *and* provides an adaptation based on the provided information quality on different service abstraction levels. The latter is a feature not offered by other matchmakers.

Finally, we want to discuss the quantitative specifics of COV4SWS.KOM: There are two major differences between common matchmakers and the work at hand. First, semantic matchmaking is usually based on subsumption matching as presented by Paolucci et al. [18]. This approach applies DoMs for discrete elements in a service description and defines the *minimum* DoM found as the overall service (or operation) DoM. This leads to quite a coarse-grained, discrete scale of possible service DoMs. In order to further rank service offers based on

Table 2. Comparison of COV4SWS.KOM with State-of-the-Art Matchmakers

Matchmakers	AP'	Q'	nDCG'
COV4SWS.KOM (Version 4b)	0.823	0.825	0.884
LOG4SWS.KOM [13, 22]	0.837	0.851	0.896
iSeM [11, 13]	0.842	0.762	0.803
SAWSDL-MX1 [12, 13]	0.747	0.767	0.839
iMatcher [8, 13]	0.764	0.784	0.855
URBE [13, 19]	0.749	0.777	0.850

a service request, additional techniques like text similarity need to be applied. In contrast, COV4SWS.KOM applies a continuous scale which allows a more fine-grained evaluation and ranking of services.

Second, we included an information quality-aware adaptation mechanism. The application of OLS in order to determine to which degree a particular service description level should influence the matchmaking results is an intuitive approach to adapt a matchmaker to a particular service domain. With linear regression analysis it is possible to determine which part(s) of a service level description should be weighted to a disproportionately small or large extent while achieving excellent evaluation results.

Summarized, the evaluation results show that adaptation based on information quality and the usage of metrics which are usually employed to determine semantic relatedness between concepts in ontologies is a promising strategy in order to improve ontology-based matchmaking results.

5 Related Work

Since the seminal paper of Paolucci et al. [18], a large number of different matchmaking approaches has been proposed. In the following, we will consider adaptive matchmakers for SAWSDL, which today provide the best results in terms of IR metrics. For a broader discussion, we refer to Klusch et al. – according to their classification, COV4SWS.KOM classifies as an adaptive and non-logic-based semantic matchmaker [9, 13].

iMatcher applies an adaptive approach to service matchmaking by learning different weightings of linguistic-based similarity measures [8, 13]. *iSeM* is an adaptive and hybrid semantic service matchmaker which combines matching of the service signature and the service specification [11]. Regarding the former, strict and approximated logical matching are applied, regarding the latter, a stateless, logical plug-in matching is deployed. In *SAWSDL-MX*, three kinds of filtering, based on logic, textual information, and structure are applied; the matchmaker adaptively learns the optimal aggregation of those measures using a given set of services [12]. Notably, COV4SWS.KOM and SAWSDL-MX/iSeM have been developed completely independently. *URBE* calculates the syntactic or semantic similarity between inputs and outputs [19]. Furthermore, the similarity between the associated XSD data types for a given pair of inputs or outputs is calculated based on predefined values. Weights may be determined manually.

In our former work, we have presented *LOG4SWS.KOM*, which is also a matchmaker for service formalisms like SAWSDL and hRESTS [14, 22]. This matchmaker shares some features with COV4SWS.KOM, especially the fallback strategy and the operations-focused matching approach.

However, LOG4SWS.KOM applies a completely different strategy to assess the similarity of service components, as the matchmaker is based on logic-based DoMs respectively their numerical equivalents. Most importantly, an automatic adaptation to different qualities of syntactic and semantic information on different service abstraction levels is not arranged for.

To the best of our knowledge, COV4SWS.KOM is the first matchmaker to apply an adaptation mechanism not aiming at the filtering but on the service description structure. Other matchmakers adapt their behavior by learning how to optimally aggregate different semantic matching filters, but are nevertheless bound to particular presumptions regarding the quality of semantic and syntactic information given on the different levels of the service description. The biggest advantage of COV4SWS.KOM is the direct adaptation to information quality of descriptions on the different service abstraction levels. This feature is so far unprecedented within service matchmaking and allows the automated adaptation and application of COV4SWS.KOM within different service domains. In contrast, other matchmakers might be only applicable in these service domains matching the needs of the matchmaker regarding the provided syntactic and semantic information on every service abstraction level.

6 Conclusion

In this paper, we proposed an information quality-aware approach to service matchmaking. Through the adaptation to different degrees of impact on single service abstraction levels, it is possible to adapt our matchmaker to different service domains. For this, we discussed the usage of similarity metrics from the field of information theory and the OLS-based adaptation of the matchmaking process regarding the quality of semantic and syntactic information on different service abstraction levels. We evaluated different versions of the corresponding matchmaker COV4SWS.KOM for SAWSDL. The combination of operations-focused matching, similarity metrics from the field of information theory, and self-adaptation based on the weights of different service abstraction levels led to top evaluation results regarding IR metrics.

Acknowledgements. This work is supported in part by the E-Finance Lab e. V., Frankfurt am Main, Germany (`www.efinancelab.de`).

References

1. Baader, F., Nutt, W.: Basic Description Logics. In: The Description Logic Handbook: Theory, Implementation and Applications, chap. 2, pp. 47–100. Cambridge University Press (2003)
2. Bellur, U., Kulkarni, R.: Improved Matchmaking Algorithm for Semantic Web Services Based on Bipartite Graph Matching. In: 2007 IEEE International Conference on Web Services. pp. 86–93 (2007)
3. Booth, D., Liu, C.K. (eds.): Web Service Description Language (WSDL) Version 2.0 Part 0: Primer. W3C Recommendation (June 2007)
4. Bourgeois, F., Lassalle, J.C.: An extension of the Munkres algorithm for the assignment problem to rectangular matrices. Communications of the ACM 14(12), 802–804 (1971)
5. Budanitsky, A., Hirst, G.: Evaluating WordNet-based Measures of Lexical Semantic Relatedness. Computational Linguistics 32(1), 13–47 (2006)

6. Farrell, J., Lausen, H. (eds.): Semantic Annotations for WSDL and XML Schema. W3C Recommendation (August 2007)
7. Gomadam, K., Verma, K., Sheth, A.P., Li, K.: Keywords, Port Types and Semantics: A Journey in the Land of Web Service Discovery. In: SWS, Processes and Applications, chap. 4, pp. 89–105. Springer-Verlag (2006)
8. Kiefer, C., Bernstein, A.: The Creation and Evaluation of iSPARQL Strategies for Matchmaking. In: 5th European Semantic Web Conference. LNCS, vol. 5021, pp. 463–477 (2008)
9. Klusch, M.: Semantic Web Service Coordination. In: Schumacher, M., Helin, H., Schuldt, H. (eds.) CASCOM: Intelligent Service Coordination in the Semantic Web, chap. 4, pp. 59–104. Birkhäuser Verlag (2008)
10. Klusch, M., Fries, B., Sycara, K.P.: OWLS-MX: A hybrid Semantic Web service matchmaker for OWL-S services. Journal of Web Semantics 7(2), 121–133 (2009)
11. Klusch, M., Kapahnke, P.: iSeM: Approximated Reasoning for Adaptive Hybrid Selection of Semantic Services. In: 7th Extended Semantic Web Conference. LNCS, vol. 6089, pp. 30–44 (2010)
12. Klusch, M., Kapahnke, P., Zinnikus, I.: Hybrid Adaptive Web Service Selection with SAWSDL-MX and WSDL-Analyzer. In: 6th European Semantic Web Conference. LNCS, vol. 5554, pp. 550–564 (2009)
13. Klusch, M., Küster, U., König-Ries, B., Leger, A., Martin, D., Paolucci, M., Bernstein, A.: 4th International Semantic Service Selection Contest – Retrieval Performance Evaluation of Matchmakers for Semantic Web Services (S3 Contest) (2010)
14. Lampe, U., Schulte, S., Siebenhaar, M., Schuller, D., Steinmetz, R.: Adaptive Matchmaking for RESTful Services based on hRESTS and MicroWSMO. In: Workshop on Enhanced Web Service Technologies (WEWST 2010). pp. 10–17 (2010)
15. Lin, D.: An Information-Theoretic Definition of Similarity. In: Fifteenth International Conference on Machine Learning. pp. 296–304 (1998)
16. Miller, G.A.: WordNet: a lexical database for English. Communications of the ACM 38(11), 39–41 (1995)
17. Mitchell, T.M.: Machine Learning. McGraw-Hill (1997)
18. Paolucci, M., Kawamura, T., Payne, T.R., Sycara, K.P.: Semantic Matching of Web Services Capabilities. In: First International Semantic Web Conference. LNCS, vol. 2342, pp. 333–347 (2002)
19. Plebani, P., Pernici, B.: URBE: Web Service Retrieval Based on Similarity Evaluation. IEEE Trans. on Knowledge and Data Engineering 21(11), 1629–1642 (2009)
20. Resnik, P.: Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Artificial Intelligence Research 11, 95–130 (1999)
21. Sakai, T., Kando, N.: On Information Retrieval Metrics designed for Evaluation with Incomplete Relevance Assessments. Information Retrieval 11(5), 447–470 (2008)
22. Schulte, S., Lampe, U., Eckert, J., Steinmetz, R.: LOG4SWS.KOM: Self-Adapting Semantic Web Service Discovery for SAWSDL. In: 2010 IEEE 6th World Congress on Services. pp. 511–518 (2010)
23. Vitvar, T., Kopecký, J., Viskova, J., Fensel, D.: WSMO-Lite Annotations for Web Services. In: 5th European Semantic Web Conference. LNCS, vol. 5021, pp. 674–689 (2008)
24. Wang, R.Y., Strong, D.M.: Beyond Accuracy: What Data Quality Means to Data Consumers. Management Information Systems 12(4), 5–33 (1996)
25. Wooldridge, J.M.: Introductory Econometrics: A Modern Approach. Thomson South-Western, 2. edn. (2003)