Similarity-Based Retrieval for Geospatial Semantic Web Services Specified using the Web Service Modeling Language (WSML-Core)

Krzysztof Janowicz

Abstract. *What prevents the Geospatial Semantic Web from taking off is not a missing architecture and protocol stack but, beside other aspects, the question of how Web services can be semi-automatically discovered and whether and to what degree they satisfy user requirements. Two approaches turned out to be useful for semantic-enabled geospatial information retrieval: subsumption reasoning and similarity measurement. However, while the former one can be applied to query service ontologies described in OWL-S or WSMO/WSML, most existing similarity theories are not able to cope with logic-based service descriptions. This chapter presents initial results on developing a directed and context-aware similarity measure that compares WSML concept descriptions for overlap and therefore supports retrieval within the upcoming Geospatial Semantic Web.*

1 Introduction & Motivation

The idea of the Web service oriented architecture (SOA) is based on the publish-find-bind pattern. To make a service available on the Internet the provider has to publish relevant metadata to a service broker. Next, a requestor can discover (find) registered services and establish a connection (bind) to them. From a syntactical point of view the SOA-Stack offers specifications for each part of the pattern: WSDL for Web service description, UDDI as a repository for description, discovery and integration and SOAP as protocol for service binding. However, to enable semi-automatic service discovery, i.e. to specify the capabilities of Web services and search queries in an unambiguous and computer-interpretable way, a semantic-enabled markup language becomes necessary. Moreover, beside this common language, a framework needs to be defined specifying which mandatory and optional metadata should be annotated. From the provider's perspective, service ontologies described using OWL-S (OWL-S 2005) or WSMO (WSMO 2005a) satisfy these requirements[1]. Both define functional and non-functional service properties, service grounding (binding) and a semantic-enabled annotation language. Although they specify what has to be said about a service, the definition of a semantic Web adequate search paradigm is out of their scope.

Over the last years of research, subsumption reasoning and similarity measurement turned out to be applicable for geospatial information retrieval. The idea be-

[1] A detailed comparison between both approaches is discussed in (WSMO 2005b); note however that it is written from the perspective of the WSMO community.

hind subsumption-based retrieval as described by Lutz & Klien (Lutz 2006) is to rearrange a queried application ontology taking a search concept into account and to return a new taxonomy in which all subconcepts of the injected search phrase satisfy the user's requirements. However, using this approach forces the user to ensure that the search concept is specified in a way that it is neither too generic (and therefore at a top level of the new hierarchy) nor too specific to get a sufficient result set. In fact the search concept is a formal description of the minimum characteristics all retrieved concepts need to share. Moreover no measurement structure is provided answering the question which of the returned concepts fits best. However, this is not necessarily a critical point, because all subconcepts at least share the requested properties. In contrast, similarity computes the degree of overlap between search and compared-to concepts and, as measurement structure, provides a (weak) order. Both characteristics turn out to be useful for information retrieval and matching scenarios. On the one hand the determination of conceptual overlap simplifies phrasing an adequate search concept and on the other hand the results are ordered by their degree of similarity to the searched concept. Similarity-based retrieval does not necessarily imply a subsumption relation between search and compared-to concepts (see Figure 1), in some cases even disjoint concepts may be similar to each other (e.g. Mother, Father). In contrast to subsumption-based retrieval, the search phrase typed into the system is not an artificial construct, but the concept the user is really looking for in the external service ontology without presuming that all returned concepts share a specific property.

In other words, the benefits similarity offers during information retrieval, i.e. to deliver a flexible degree of conceptual overlap to a searched concept, stand against shortcomings during the usage of the retrieved information, namely that the results do not necessarily fit the user's requirements. To make the difference between both approaches more evident one can imagine a search phrase specified by using a shared vocabulary (see Figure 1) to retrieve all concepts whose instances *overlap* with waterways. In contrast to the subsumption-based approach, similarity measurement would additionally deliver concepts whose instances are located *inside* and *adjacent* to waterways, and indicate through a lesser degree of similarity that these concepts are close to, but not identical with the user's intended concept.

Following the above argumentation, similarity supports users during information retrieval; however this presumes that the chosen similarity measure supports the representation language of the inspected service (ontology). It turns out that, besides the fact that several similarity theories make fundamentally different assumptions about how and what is measured (e.g. feature vs. geometric model (Goldstone 2005)), most of them come with their own proprietary knowledge representation format. In contrast, the majority of service ontologies are specified using standardized or commonly agreed upon logic-based knowledge representation languages and especially various kinds of description logics. This leads to a gap between available similarity theories and existing ontologies which oppose a wider application of similarity measures as part of the Geospatial Semantic Web.

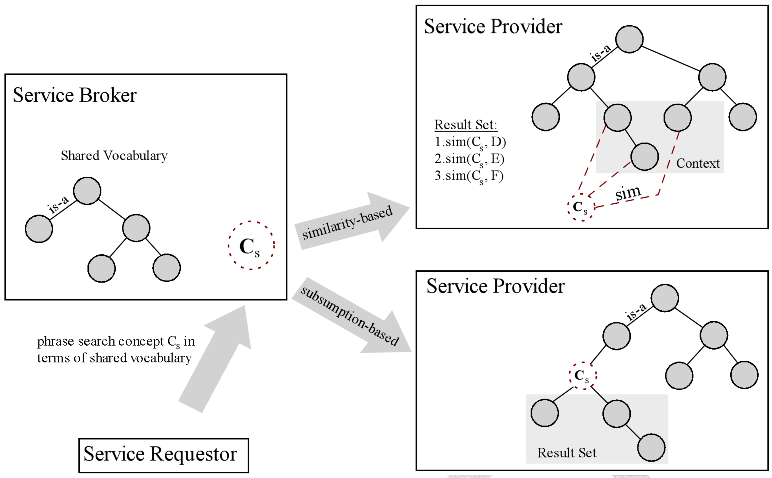

Figure 1. Subsumption and similarity-based retrieval using a shared vocabulary

Additionally, most proprietary knowledge representation formats associated with existing similarity theories lack of a formal semantics and also language constructs proven to be useful for conceptualization (such as relation-filler pairs). This is a crucial issue because in computer science the concepts between which similarity is measured are representations of the concepts in our minds. Consequently, the lack of a precise and expressive representation language has impact on the quality of the resulting similarity assessments as discussed in (Janowicz 2005) for the feature-based MDSM theory (Rodriguez 2004). The same arguments hold for geometric approaches to similarity based on Gärdenfors' idea of conceptual spaces (Gärdenfors 2000). To integrate relations and hence improve the expressivity of conceptual spaces for similarity measures, Schwering (Schwering 2005) for instance combines the geometric approach with classical network models. Initial approaches towards similarity measures for expressive description logics are discussed in (d'Amato 2005; Janowicz 2006). A theory applying similarity for Web service comparison based on OWL-S is presented in (Hau 2005); however it does not take neighborhood and alignment models into account. An overview about existing similarity theories, their application areas and characteristics is out of the scope of this chapter and was recently discussed in (Goldstone 2004).

The chapter presents initial results on how similarity measurement can support semi-automatic information retrieval and matching tasks within the upcoming Geospatial Semantic Web.

2 Similarity between WSML Concept Descriptions

This section describes the proposed similarity measurement framework focusing especially on attribute -filler (respectively relation-filler) similarity. Starting with a service integration scenario the used representation language (WSML-Core) will be introduced and the similarity framework will be discussed step by step.

2.1 Scenario

A European lodging portal on the Internet is providing information about accommodations in cities attractive to tourists. To avoid maintenance costs the service provider does not store the information in a local database, but dynamically connects to external Web services. However, to offer a consistent interface and vocabulary to the portal users the service provides its own terminology. To do so, the types of accommodations distinguished in the external services have to be aligned to the local terminology. One of the external services, delivering information about accommodations in Amsterdam, provides separate conceptualizations for houseboats and botels[2] while the local ontology does not make this distinction. The task of similarity measurement within this scenario is to propose whether botels should be displayed as houseboats, hotels or youth hostels within the local terminology presented via a Web interface to the user. The provider therefore runs a similarity query against the local ontology using the external concept Botel as search phrase (C_s). In addition, the service provider specifies a search context, i.e. a description of the minimum requirements all compared-to concepts need to fulfill (to be housings in this case). The result of the query is a list of similarity values indicating how close the compared conceptualizations are. It is assumed that both, the external service and the accommodation portal, stick to a shared vocabulary (Figure 1) that specifies the base concepts of the domain and that the concepts Botel, Houseboat, Hotel and Youth_Hostel are defined in terms of this shared vocabulary.

2.2 WSMO and WSML

As similarity between concepts is based on their specification and the chosen representation language, this section gives a brief overview about the Web Service Modeling Language (WSML) and introduces simplified conceptualizations for the types of accommodations distinguished in the scenario.

Based on the Web Service Modeling Framework (WSMF) developed by Fensel and Bussler (Fensel 2002), the Web Service Modeling Ontology (WSMO) specifies four main modeling elements describing various aspects of semantic Web services needed within the publish-find-bind pattern.

- **Ontologies** providing the formal semantics for goals, Web services and mediators and linking human and machine terminology together.
- **Goals** specifying the user's aims with respect to the requested service functionalities
- **Web services** representing the offered functionality in terms of its capabilities and non-functional properties.
- **Mediators** offering several types of mediators to overcome interoperability problems.

While WSMO describes what needs to be said, WSML (WSML 2005) is the corresponding modeling language providing a formal syntax and semantics to describe these elements in a machine-interpretable and unambiguous way. It supports both a

[2] For instance: Hotel Amstel Botel Amsterdam: http://www.amstelbotel.nl/

condensed machine oriented as well as a human readable syntax and comes in five flavors of different expressivity: WSML-Core, WSML-DL, WSML-Flight, WSML-Rule and WSML-Full. Independent from a certain variant, WSML distinguishes between following language elements: concepts (and their attributes), relations, instances (of concepts and relations) and axioms. However, the abilities to describe them depend on the chosen WMSL flavor. For each element, additional non-functional properties, mostly taken from the Dublin Core schema, can be specified.

The similarity measurement framework introduced in this chapter is defined for the WSML-Core variant that is based on the intersection of description logics with logic programming and acts as a base and exchange vocabulary for WSMO. In WSML-Core the usage of relations is restricted to binary predicates and cardinality restrictions are not supported. The WSML documentation recommends using concept attributes instead of relations wherever possible. Moreover WSML-Core does not allow for specifying the attribute features transitive, symmetric, reflexive and inverseOf within local concept descriptions. However, they can be added as global axioms to the service ontology and linked to the intended concept via the Dublin Core element dc:relation (WSML 2005; p. 27). Although WSML distinguishes between constraining (ofType) and inferring (impliesType) attribute and relation descriptions, the former can only be applied to datatypes within the Core variant (WSML 2005; p. 17). WSML offers built-in datatypes, such as strings, integers, doubles or dates which correspond to XML Schema datatypes and operators (XQuery functions) such as equal or numericGreaterThan. The syntax and semantics (mapped to Horn Logic) of the language constructs used within WSML-Core as well as an exemplary concept definition are depicted in Table 1. See (WSML 2005; p. 27-30) for further details.

Table 1. Syntax and Semantics of WSML-Core

WSML-Core (syntax)	Horn Logic (semantics)	Example
π(*head* **impliedBy** *body*.)	π(*head*) ← π(*body*)	Youth_Hostel **subConceptOf** {Building, Housing} **nonFunctionalProperties** dc#description **hasValue** 'concept of a youth hostel' category **ofType** _integer service **impliesType** SelfService offers **impliesType** Room
π(*lexpr* **or** *rexpr*)	π(*lexpr*) ∨ π(*rexpr*)	
π(*lexpr* **and** *rexpr*)	π(*lexpr*) ∧ π(*rexpr*)	
π(*X1* **memberOf** *id2*)	*id2*(*X1*)	
π(*id1* **subConceptOf** *id2*)	*id2*(x) ← *id1*(x)	
π(*X1*[*id2* **hasValue** *X2*])	*id2*(*X1*,*X2*)	
π(*id1*[*id2* **impliesType** *id3*])	*id3*(y) ← *id1*(x) ∧ *id2*(x,y)	
π(*id1*[*id2* **ofType** *dt*])	*dt*(y) ← *id1*(x) ∧ *id2*(x,y)	
$\pi(p(X_1,...,X_n))$	$p(X_1,...,X_n)$	

Although we stick to the human readable syntax within this chapter, it has to be mentioned that compared WSML concepts have to be preprocessed before similarity is measured. The necessary steps are described in (WSML 2005; p.42f) and result in a WSML normal form (see also (Janowicz 2006)). The underlying idea is to decompose complex descriptions to simple ones. Note that concepts inherit all attributes specified for their ancestors.

Table 2 shows possible conceptualizations for the types of accommodations described in the scenario. In contrast to Hotel and Youth_Hostel (see Table 1), Botel and Houseboat are defined as subconcepts of Boat; however houseboats are usually self serviced and are rented as a whole and not per room.

Table 2. Conceptualizations for the Botel-Houseboat scenario

Botel	Houseboat	Hotel
subConceptOf {Boat, Housing} category **ofType** _integer service **impliesType** Service offers **impliesType** Room borders(i)[3] **impliesType** Waterway	**subConceptOf** {Boat, Housing} category **ofType** _string service **impliesType** Self-Service inside **impliesType** Waterway	**subConceptOf** {Building, Housing} category **ofType** _integer service **impliesType** Service offers **impliesType** Room

2.3 Similarity Measurement Framework

The presented theory measures similarity between concepts (in normal form) by stepwise comparing their WSML-Core descriptions, where a high level of overlap indicates high similarity and vice versa. To do so, all available language constructors, i.e. subConceptOf / subRelationOf and attribute (respectively relation) as well as the restrictions for their fillers by ofType and impliesType, have to be taken into account. Similarity (sim) is therefore defined as a polymorphic, binary and real-valued function $\mathbf{X} \times \mathbf{Y} \rightarrow \mathbb{R}\ [0,1]$ providing implementations for all language constructs. The overall similarity (sim_o) between concepts is just the normalized (and weighted) sum of the single similarities calculated for all compared-to parts of the concept descriptions. A similarity value of 1 indicates that compared concepts are equal, whereas 0 implies total dissimilarity. In the following σ denotes the normalization factor while ω is used to represent weightings.

In general a similarity measurement framework consists of the following five phases - their concrete implementation and relative importance however depends on the chosen representation language.

- Define search concept and context
- Generate canonical normal form for compared concepts
- Align parts for comparison
- Apply similarity functions to compared-to parts
- Derive normalized overall similarity

Preprocessing steps to derive a WSML-Core normal form (phase 2) have been discussed in section 2.2 and are therefore not considered here in further detail. A more complex example pointing out the importance of canonical representation for similarity measurement is discussed in (Janowicz 2006).

[3] Note that *borders(i)* (borders from inside) corresponds to TPP and *inside* to NTTP in RCC8 (Cohn 1997); however these relations need more investigation for 3D spatial neighborhoods (Kuhn 2002).

2.3.1. Search Concept and Context

As depicted in Figure 1, a search concept (also called source) is phrased in terms of a shared vocabulary and compared to the concepts (called targets) in an examined ontology. The search concept needs to be specified in the same representation language as the target concepts or mappings between the languages have to be defined. The target concepts are not necessarily just all concepts in the examined ontology but defined by a search context. The idea of context (see also the Matching Distance Similarity Measure MDSM (Rodriguez 2004)) is on the one hand to determine which parts from the service ontology have to be compared to the search concept and on the other hand to influence the measured similarity making it situation-aware. Within the presented approach context is used to combine the benefits of subsumption reasoning and similarity-based retrieval. It is defined as a set of concepts from the examined application ontology that, after reclassification (comparable to the Lutz & Klien approach (Lutz 2004)), are subconcepts of $C_{lcs:}$ Context = $\{C|\ C \sqsubseteq C_{lcs}\}$. In other words, context determines the universe of discourse (called application domain in (Rodriguez 2004)). In the presented accommodation scenario, C_{lcs} guarantees that all concepts proposed to be similar to Botel at least act as accommodations (subconcepts of Housing). Therefore similarity to cargo ships or ferries would not be measured, although they are kinds of boats as well.

2.3.2. Alignment Matrix

After their expansion to WSML-Core normal form, concepts are lists of attributes respectively relations (with restrictions for their fillers), including also those inherited from their ancestors. While search concept and context define which concepts are compared to each other, next it has to be determined which parts (e.g. which attribute-filler pairs) of the selected concepts are compared to each other. To do so, a matrix $C_s \times C_t$ of all possible combinations is generated. Similarity can only be computed between the same kind of language elements, i.e. attribute-filler pairs using the ofType keyword are not compared to those using impliesType and so on. Therefore such pairs are not further taken into account. Next, the following steps are applied for all parts of the source concept description and each parts of C_s and C_t are only selected once:

- If the matrix contains an identical attribute/relation-filler pair for the search and the target concept, the similarity for this pair is 1 and the normalization factor σ is increased by 1.
- If the matrix contains an attribute/relation-filler pair out of the target concept description where the attribute/relation is identical to the pair in the source concept but the fillers are different, similarity between the fillers is calculated. If there are more such pairs, the one with the highest similarity for the filler is selected and σ is increased by 1.
- If for an attribute/relation-filler pair out of the source concept description no pair with an identical attribute/relation could be found, the most similar pair is selected where the similarity between the attributes/relations can be determined using a conceptual neighborhood graph; σ is increased by 1.

- If no neighborhood graph is specified for the compared-to attributes/relations, their similarity is measured by co-occurrence and σ is increased by 1.
- Co-occurrence is also determined for superconcepts if they are base symbols (primitives) defined in the shared vocabulary and therefore have no description to be compared (inherited); σ is increased by 1.
- For parts of the search concept that could not be compared similarity is 0 and σ is increased by 1; while σ is not increased for parts of the target concept that could not be aligned to corresponding parts of the search concept.

In other words, for each part of the search concept's description, a counterpart from the compared-to concept's description is chosen in a way that a meaningful similarity can be computed between them afterwards and each part is only examined once. The alignment phase is directed, i.e. asymmetric (Rodriguez 2004), in a sense that the resulting overall similarity depends on the search direction. Therefore $sim_o(C_s, C_t)$ is not necessarily equal to $sim_o(C_t,C_s)$. While each element of the search concept's description is compared to an element from the compared-to concept, some parts of the latter may remain uncompared. This is always the case if the target concept is specified by more elements than the search concept. The similarity value for these remaining parts is always 0 while they do not increase the normalization factor σ. If however the search concept is described by more elements than can be compared, σ is increased by 1 for each remaining part. As a result the overall similarity is decreased. In other words, if the target concept in the examined ontology is more specific than requested by the user (via the search concept) this has no impact on the measured overall similarity. On the other side, similarity decreases if the user's search concept is more specific than its counterpart in the examined ontology.

2.3.3. Similarity Functions

After determining which parts of the search and target concept are compared to each other, the similarity between these selected parts is calculated. Two situations have to be taken into account here: the similarity between attribute/relation-filler pairs depends on both, the similarity between the attributes (respectively relations) and the similarity between the fillers. If the fillers are concepts again, the similarity framework is recursively applied to the compared fillers, i.e. an alignment matrix is created for the compared concepts and similarity for the selected parts is measured. If, however, the fillers are datatypes (expressed via the keyword ofType in WSML-Core), similarity is determined via a matching function and no recursion is necessary.

$$sim_{cf}(ac_s, ac_t) = sim_a(a_s, a_t) * sim_o(c_s, c_t) \qquad \textbf{(1)}$$

Equation 1 shows how the attribute-filler[4] similarity (sim_{cf}) is calculated for concept fillers, where sim_a is the similarity between attributes and sim_o is the overall similarity between their fillers. In addition to this multiplicative approach similarity could also be defined as (weighted) average between attribute and filler similarity, which is discussed in section 3.

According to the alignment matrix similarity between attributes (sim_a) can be determined in two ways: using a conceptual neighborhood graph (ndw), as depicted in Figure 2 for topological neighborhood, or via co-occurrence (sim_{co}). The benefit of conceptual neighborhoods is that they imply a very natural notion of similarity - which is just the inverse and normalized graph distance between the compared attributes (see Equation 2). Although the edge weightings may vary with respect to the chosen conceptual neighborhood (n) or intersection matrix, they are usually set to 1 per edge and symmetric (see also (Bruns 1996; Li 2006) for similarity measures between spatial scenes).

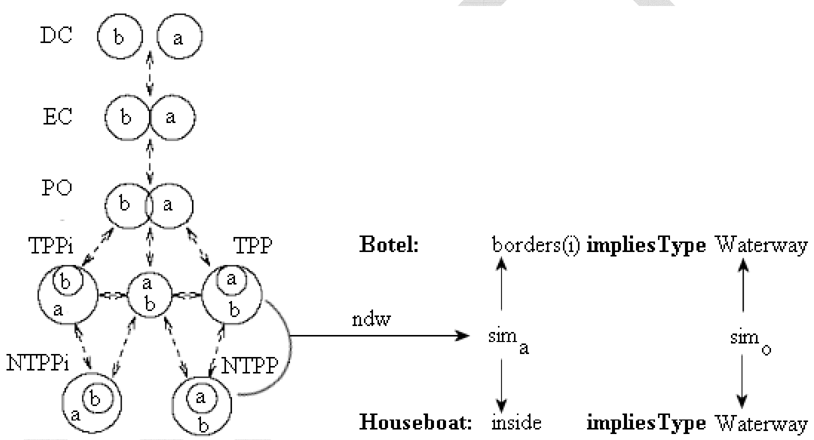

Figure 2. Spatial neighborhood distance (Cohn 1997) and inter-attribute similarity

Equation 2 describes how the similarity between attributes is determined via their relative distance within a conceptual neighborhood, where $distance_n$ between a_s and a_t is the shortest path trough the graph while $max\_distance_n$ represent the longest path.

$$ndw(a_s,a_t)=\frac{max\_distance_n - distance_n(a_s,a_t)}{max\_distance_n} \quad \textbf{(2)}$$

In contrast the co-occurrence (also called common subsumee approach) assumes that attributes are more similar, if they share more common sub-attributes. In Equation 3, sim_{co} is defined as the ratio between the number of subsumees of both attributes and the number of sub-attributes of one or both of them (and all y are elements of the context). This notion of co-occurrence is comparable to Jaccard similarity

[4] Similarity for relations-filler pairs is calculated accordingly, but is omitted here for reasons of readability.

coefficient (Tan 2005). Note that within WSML-Core the sub-attribute relationship is specified as implication using logical expressions (WSML 2005; p.29). The letters x and y are chosen here, to indicate that the same equation is applied to attributes, relations and primitive concepts (base symbols) as well.

$$sim_{co}(x_s, x_t) = \frac{|\{y \mid (y \sqsubset x_s) \sqcap (y \sqsubset x_t)\}|}{|\{y \mid (y \sqsubset x_s) \sqcup (y \sqsubset x_t)\}|} \quad \textbf{(3)}$$

As can be seen from Equation 1, the attribute-filler similarity sim_{cf} calls the overall similarity sim_o to determine the overlap between involved concept fillers. The overall similarity, however, again invokes sim_a to compare the attributes specified for these concepts and so on. The process terminates when the concepts specified as fillers have no concept description, i.e. are base symbols (primitives) of the shared vocabulary. According to section 2.3.2, their similarity is determined via Equation 3, where the subsumees are not attributes but subconcepts. The same approach is applied if super concepts defined in the head of concept definitions are base symbols and therefore do not bequeath attributes to their subconcepts.

While the former paragraphs focused on concepts as fillers, the similarity (sim_{df}) between attribute-filler pairs with datatype fillers is determined according to Equation 4. The function match() returns 1 for the same type or if all instances of d_s could be converted to d_t without losing information (respectively precision; such as from integers to decimals (WSML 2005; p.88)); otherwise match() returns 0. Some problems related to similarity with respect to datatypes are discussed in the further work section.

$$sim_{df}(ad_s, ad_t) = sim_a(a_s, a_t) * match(d_s, d_t) \quad \textbf{(4)}$$

2.3.4. Overall Similarity

Finally, the overall similarity (sim_o) between search and target concept is the normalized sum of the similarities derived by comparing attributes with concept fillers (via sim_{cf}), attributes with datatype fillers (via sim_{df}) and primitive concepts (base symbols) in the head of c_s and c_t (via sim_{co}). In Equation 5, (c_{si}, c_{tj}) represents the parts of the source and target concept selected for comparison within the alignment matrix AM_{st}.

$$sim_o(c_s, c_t) = \frac{1}{\sigma} \sum_{(c_{s_i}, c_{t_j}) \in AM_{st}} sim(c_{s_i}, c_{t_j}) \quad \textbf{(5)}$$

3 Human Subject Testing

This section describes the results from a Web-based human subject test, developed to examine how users rate the similarity between attribute/relation-filler pairs. After explaining the goals of the test, subjects were asked to make similarity estimations using a slider that ranges from very dissimilar to very similar (which corresponds to a value range between 0 and 100). The slider was situated between the compared entities and its start position was on half way between both. The test consists of three steps, each containing four pairs to be compared. In the first step subjects were asked to rate similarity between spatial relations such as *disjoint* and *meets*. In the second step object pairs such as *waterway* and *river* were compared. Finally, in the third step subjects had to rate the similarity between combinations of both (e.g. *disjoint waterway – meets river*).These similarity estimations were than compared to automatically generated similarity values using three different approaches: the average, a weighted average with flexible weightings and the multiplicative approach depicted in Equation 1. The necessary attribute and filler similarities were taken from the first two steps of the test.

Out of 84 similarity estimations derived from step three, 80 were taken for further computation. As depicted in Figure 3, the multiplicative approach produces the best results. In 41 out of 80 cases the absolute deviation does not exceed 10 points; however the approach tends to underestimate in general. In contrast, the weighted average tends to overestimate and the results are not as precise (33 out of 81). The simple average approach was always overestimating and the deviation from human's estimations was high in general.

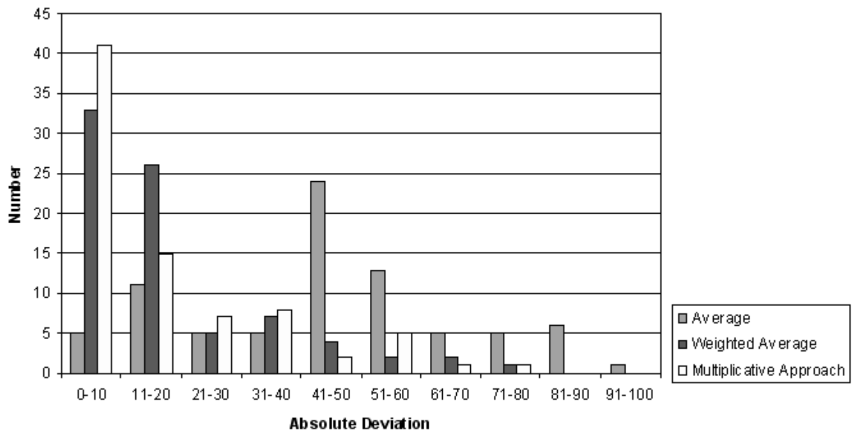

Figure 3. Absolute deviation between machine and human similarity estimations

To explain the idea of similarity estimations to the subjects, they were told that comparing relation-object pairs could be imagined as rating how probable two people (describing a certain situation in different words) actually talk about the same situation or not. It turns out, that this explanation may be a reason why some of the human's similarity estimations were inconsistent and neither captured by the multiplicative approach nor the weighted average While *inside - disjoint* and *lake – channel* were rated to be dissimilar, the combination was rated to be more similar than expected. Subjects assumed that if a described object is inside a lake, it is disjoint from a channel.

4 Discussion and Further Work

The directed and context-aware similarity theory presented within this chapter is able to measure the overlap between concepts specified using WSML-Core, and can therefore support integration and retrieval within service oriented architectures. In contrast to previous work, it points out possible ways of combining subsumption reasoning and similarity. Nevertheless, a lot of work remains to be done to apply these initial results to sophisticated real world applications.

Referring to the accommodation scenario, it turns out that botels are more similar to hotels (0.67) than to houseboats (0.62) or youth hostels (0.5). However, the measured similarities depend on the representation of the compared concepts within the provider's ontology. Services focusing on vessels instead of accommodations may use different conceptualizations, making Botel and Houseboat more similar. Note that from now on the accommodation service can also display botels on the portal's Website whenever a user is looking for hotels in Amsterdam, but (in contrast to subsumption-based retrieval) integrating the concept Botel into the local knowledge base would lead to inconsistencies (a botel is not a building).

It turns out, that while the comparison of attributes (respectively relations) restricted by concept fillers is well examined (d'Amato 2005; Schwering 2005; Janowicz 2006), the question of how to develop a meaningful theory for datatype similarity still remains unsolved. One of the main reasons is missing information about the level of measurement or non-linear measures (Schade 2005). For instance, the category of a hotel is measured in stars and represented as an integer on an ordinal scale; while the distance to a beach is also of the datatype integer but on an interval scale: 100 meters to the beach is half as much as 200 meters, but a 2 star hotel is not half as good as a 4 star hotel. In addition, according to Equation 4, the match function returns 0 for comparing decimals to integers, although the lost precision may not be relevant for a user in a certain situation. Taking complex XSD types into account would further complicate the determination of a meaningful notion of datatype similarity (e.g. xs:sequence).

Another important issue is the extension of the presented approach to cope with more expressive WSML variants. The major question arising here is what can be said (in terms of similarity) about compared logical expressions (e.g. via generalization). While the presented theory demonstrates how to compare concepts within WSML service ontologies, mediators, goals and capabilities were not discussed within this approach. However further theories may benefit from the idea of WSML mediators as mapping rules (WSMO 2005a). Moreover it has to be examined how users, such as the service provider, can phrase search concepts without being domain experts and trained logicians. Finally, further refined human subject tests are necessary.

References

Bruns, T.H. and M.J. Egenhofer (1996). Similarity of Spatial Scenes, in Seventh International Symposium on Spatial Data Handling (SDH '96), M.-J. Kraak and M. Molenaar, Editors.Delft, Netherlands: p. 31-42.

Cohn, A.G., et al., (1997). Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. GeoInformatica. 1(3): p. 275-316.

d'Amato, C., N. Fanizzi, and F. Esposito. (2005). A Semantic Dissimilarity Measure for Concept Descriptions in Ontological Knowledge Bases in The Second International Workshop on Knowledge Discovery and Ontologies. Porto, Portugal.

Fensel, D. and C. Bussler. (2002). The Web Service Modeling Framework WSMF. Electronic Commerce Research and Applications. 1(2).

Gärdenfors, P. (2000) Conceptual Spaces - The Geometry of Thought. Cambridge, MA: Bradford Books, MIT Press.

Goldstone, R. L, and J. Son (2005). Similarity. In K. Holyoak & R. Morrison (Eds.). Cambridge Handbook of Thinking and Reasoning. Cambridge: Cambridge University Press: p. 13-36.

Hau, J., W. Lee, and J. Darlington. (2005). A Semantic Similarity Measure for Semantic Web Services in Web Service Semantics Workshop 05 at WWW2005. Chiba, Japan.

Janowicz, K. (2005). Extending Semantic Similarity Measurement by Thematic Roles, in First International Conference on GeoSpatial Semantics, GeoS 2005, Mexico City. Springer Verlag: Berlin: p. 137-152.

Janowicz, K. (2006). Sim-DL: Towards a Semantic Similarity Measurement Theory for the Description Logic ALCNR in Geographic Information Retrieval. R. Meersman, Z. Tari, P. Herrero et al. (Eds.): SeBGIS 2006, OTM Workshops 2006, LNCS 4278: p. 1681 – 1692.

Kuhn, W. (2002). Modeling the Semantics of Geographic Categories through Conceptual Integration. in Geographic Information Science-Second International Conference (GIScience 2002, Boulder, Colorado).Springer-Verlag, Lecture Notes in Computer Science 2478.

Li, B. and F.T. Fonseca. (2006). TDD - A Comprehensive Model for Qualitative Spatial Similarity Assessment. Spatial Cognition and Computation, 6(1): p. 31-62.

Lutz, M. and E. Klien (2006). Ontology-Based Retrieval of Geographic Information. International Journal of Geographical Information Science. 20(3): p. 233-260.

OWL-S (2005). OWL-S 1.0 Release: www.daml.org/services/owl-s/1.0/. {09-14-2006}.

Rodríguez, A.M. and M.J. Egenhofer (2004). Comparing Geospatial Entity Classes: An Asymmetric and Context-Dependent Similarity Measure. International Journal of Geographical Information Science. 18(3): p. 229-256.

Schade, S. (2005). Sensors on the Way to Semantic Interoperability. Proceedings GI-Days 2005: Geo-Sensor Networks - From Science to Practical Applications. ifgi-Prints 23.

Schwering, A. (2005). Hybrid Model for Semantic Similarity Measurement. in 4th International Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE05). Agia Napa, Cyprus. Springer Verlag.

P.-N. Tan, M. Steinbach, and V. Kumar. (2005). Introduction to Data Mining, Addison Wesley Publishing Company .

WSML (2005). The Web Service Modeling Language WSML (D16.1v0.21): www.wsmo.org/TR/d16/d16.1/v0.21/ {09-14-2006}.

WSMO (2005a). WSMO Primer (D3.1v0.1): www.wsmo.org/TR/d3/d3.1/ {09-14-2006}.

WSMO (2005b). A Conceptual Comparison between WSMO and OWL-S (D4.1v0.1): www.wsmo.org/TR/d4/d4.1/v0.1/. {09-14-2006}.