International Journal of Innovative Computing, Information and Control
Volume **13**, Number **4**, August **2017**

ICIC International ©2017 ISSN 1349-4198
pp. 1221–1242

A HYBRID SEMANTIC SIMILARITY MEASURING APPROACH FOR ANNOTATING WSDL DOCUMENTS WITH ONTOLOGY CONCEPTS

Wei Lu[1], Yong Yang[1,*], Weiwei Xing[1], Xiaoping Che[1]
Yuanyuan Cai[1] and Liqiang Wang[2]

[1]School of Software Engineering
Beijing Jiaotong University
No. 3, Shangyuancun, Haidian District, Beijing 100044, P. R. China
*Corresponding author: 12112088@bjtu.edu.cn

[2]Department of Computer Science
University of Central Florida
Orlando, FL 32816, USA

Received February 2017; revised May 2017

Abstract. *Semantic annotation of legacy Web Services is one of the fast and efficient ways to implement Semantic Web Service paradigm. Semantic similarity between concepts in WSDL (Web Services Description Language) document and ontology concepts is the backbone of semantic annotation of legacy Web Services. The overwhelming majority of previous works focused mainly on semantic similarity of concepts in a specific domain ontology. However, the concepts used in Web Services are often from multiple sources or different domain ontologies. This makes traditional approaches no longer applicable. To address this, we propose a hybrid measuring approach to measure semantic similarity between concepts in WSDL documents and concepts in OWL (Ontology Web Language) files. The proposed approach mainly consists of two parts: lexical-level similarity measuring and structural-level similarity measuring. Specially, we fusion adopt three commonly used approaches, i.e., edge-based, feature-based, and information content-based semantic similarity measuring approaches. Specifically, we map the above mentioned three approaches to three proposed internal features, i.e., depth, width, and density, in the abstract tree structure when measuring structural-level similarity. We conduct experimental comparisons, and the results show that the proposed approach provides better accuracy. Furthermore, the proposed approach can be applied in any user defined Web Services description documents in theory with a wider range of application.*

Keywords: Semantic similarity, Semantic annotation, Web Services, Semantic Web Services, Ontology

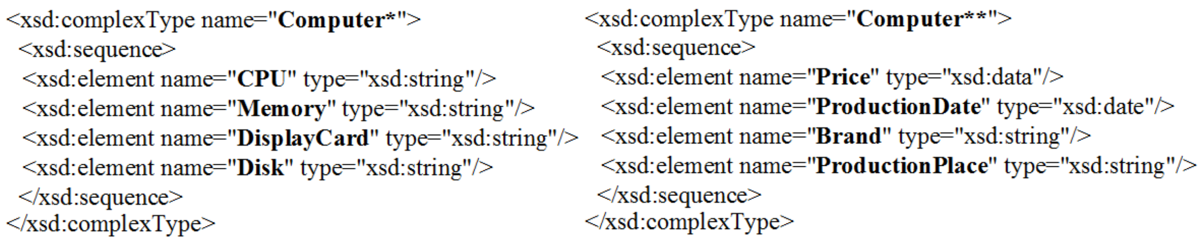
1. **Introduction.** The technology of Web Services is independent of underlying implementation technologies and platforms [1]. It provides a standardized way to achieve inter-operability between heterogeneous software systems. However, it is difficult to discover and compose relevant Web Services when the amount of services increases rapidly and data heterogeneity continues to grow. In the meanwhile, the accuracy of Web Services discovery and composition decreases because the syntactic-based description of Web Services lacks semantics [2].

Semantic Web Services can provide high accuracy for Web Service discovery, and facilitate composition significantly. Furthermore, reusability of Web Services can be improved in the meanwhile [5]. One realistic and effective way to implement Semantic Web Services is annotating legacy Web Services with concepts that are from formal logic-based ontologies. In this case, intelligent agents and service-based applications can actually reason on the formal service semantics from an AI (Artificial Intelligence) perspective [3, 4].

The core idea of semantic annotation is tagging a concept in a WSDL document with a proper domain ontology concept. Because ontologies (that formally represent knowledge as a set of concepts with relationships) can provide a definitive and exhaustive classification of entities in all spheres of being, semantic similarity, that reflects how closely associated concept pairs is, is the backbone of semantic annotation. According to the result of semantic similarity measuring, concepts in WSDL documents can be annotated with selected ontology concepts.

At the beginning of the research, researchers focus mainly on semantic similarity measuring between concept pairs in a single ontology. Approaches that have been proposed to measure semantic similarity between concept pairs in a single ontology can be roughly classified as: edge-based [8, 9, 10, 11], feature-based [6, 7, 12, 13, 14] and information content-based [15, 16, 17, 18]. With the widespread adoption of the Semantic Web paradigms, many ontologies have been developed in the past few decades for various purposes and domains [19]. Joint application of multiple ontologies has been considered when knowledge in single ontology is not enough [20]. Many works have been done in semantic similarity measuring in multiple ontologies [13, 21, 22, 23, 24].

However, WSDL documents are often written manually by different program developers at the present stage. More seriously, different developers define the same concept from different scopes, points of view perspectives, and design principles. In addition, different concepts may be defined from a similar perspective with the same semantic. Take *"Computer"* in WSDL document fragment in Figure 1 for example.



```
<xsd:complexType name="Computer*">
  <xsd:sequence>
   <xsd:element name="CPU" type="xsd:string"/>
   <xsd:element name="Memory" type="xsd:string"/>
   <xsd:element name="DisplayCard" type="xsd:string"/>
   <xsd:element name="Disk" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="Computer**">
  <xsd:sequence>
   <xsd:element name="Price" type="xsd:data"/>
   <xsd:element name="ProductionDate" type="xsd:date"/>
   <xsd:element name="Brand" type="xsd:string"/>
   <xsd:element name="ProductionPlace" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

FIGURE 1. Example definitions of concept "Computer" by different developers

The meaning of "*Computer**" and "*Computer***" represents the same object (i.e., a computer machine). However, the definition of *Computer** may be used in a scenario that performance is critical when "*Computer***" focuses mainly on price.

This is a common issue in Web Services that different service providers provide similar services. However, these services may focus on different perspectives that lead to the definition of the same concept from different perspectives. There are problems when directly applying traditional semantic similarity measuring approaches in single or multiple ontologies in this scenario. Because these approaches are difficult to detect this subtle semantic differences or they will amplify this difference to much. Traditional semantic similarity measuring approaches have limitations when concepts are described by different language (such as XML, RDF, OIL, or OWL). Furthermore, those approaches are inflexible because most of them are proposed to measure semantic similarity in a single ontology.

In this paper, we propose a hybrid semantic similarity measuring approach to measure semantic similarity between two concepts that are in WSDL and OWL documents respectively. The proposed approach bases on a fact that semantic similarity between concepts is heavily influenced by relevant concepts in the same document.

In the proposed approach, we use an abstract tree structure (consists of named nodes and undirected edges) as an intermediate expression to represent concepts and the relationships (such as inheritance relationship) between the concepts in WSDL and OWL documents. Based on the tree structure, the proposed approach combines the lexical-level and the structural-level similarity to measure semantic similarity. The lexical-level similarity is the measuring of the linguistic similarity between two concepts based on their names. It just considers the similarity of appearance between concept pairs. The structural-level similarity is the measuring of the structural similarity between the concept pairs with considering the relevant concepts in the same document of the compared concept pairs, because the relevant concepts of the compared concept pairs influence the semantic similarity of the concept pairs. The proposed approach adopts Levenstein Distance [33] and Abbreviation Expansion [38] to measure lexical-level similarity. It considers three kinds of internal features of nodes in the tree structure when conducting structural-level similarity measuring. Three commonly used approaches, i.e., edge-based, feature-based, and information content-based semantic similarity measuring approaches, are mapped to three proposed internal features, i.e., depth, width, and density, of nodes in the tree structure. Finally, the semantic similarity is measured through comprehensive consideration of both lexical-level and structural-level similarity. Dynamic weights of lexical-level and the structural-level similarity are used to reveal the contribution of the lexical-level and structural-level similarity to the final semantic similarity.

The proposed hybrid semantic similarity measuring approach has some contributions as:

(1) Semantic similarity is divided into two parts (lexical-level and structural-level similarity). It comprehensively considers both the appearance of the concept's name and the relevant concepts with relationships of the compared concept pairs. Therefore, it can avoid the limitations of separately considering of concept's name or node's structure attribute in the tree structure;

(2) Three internal features (i.e., depth, width, and density) of a node in the tree structure are considered when measuring structural-level similarity. This approach makes full use of the advantages of three existing commonly used approaches (i.e., edge-based, feature-based, and information content-based semantic similarity measuring approaches).

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 describes the proposed approach. Section 4 gives the experimental results. Sections 5 is conclusion and future work.

2. **Related Work.** The semantic similarity has been extensively applied in various applications, e.g., information extraction and retrieval, data privacy, sense disambiguation and clustering, and Semantic Web discovery. In the past decades, many ontology-based semantic similarity measuring approaches have been proposed. In this section, we review existing works of semantic similarity measuring approaches in a single ontology at first. Then, we give reviews of semantic similarity measuring approaches in multiple ontologies. At last, we represent some works that are related to this work.

2.1. **Semantic similarity in a single ontology.** Existing works of semantic similarity measuring in a single ontology can be roughly classified into edge-based, feature-based and information content-based.

Edge-based approaches. This kind of measuring approaches takes an ontology as a directed graph in which semantic distance represents the semantic similarity between two concepts. The semantic distance is the number of links separating the concept pairs [8, 9, 10, 11]. It is the shortest path of all possible paths between two concepts "a" and "b" as Formula (1):

$$Distance(a,b) = \min_{\forall i} |path(a,b)| \tag{1}$$

Edge-based approach chooses the least path distance between nodes in the directed graph as the semantic distance. Then, a function $F()$ is used to measure the semantic similarity based on the semantic distance as Formula (2):

$$Sim(a,b) = F(Distance(a,b)) \tag{2}$$

Different approach has different function of $F()$. The core idea of $F()$ normalizes the semantic similarity in the range of $[0,1]$.

However, the result of this approach is influenced by the appearance of homonyms. Besides that, edge-based approach takes the relative distance between node pairs into account without considering the absolute depth of the node. This is contrary to one of the implicit ontology design principles (concepts with abstract meaning should be in the upper layer when concepts with concrete meaning should be in the lower layer).

Feature-based approaches. These measuring approaches utilize concept's feature sets when measuring semantic similarity [6, 7, 12, 13, 14]. They measure the semantic similarity of concept pairs "a" and "b" by a function of features (e.g., concept's ancestors, and concept's descriptions) overlapping set and non-overlapping set as Formula (3):

$$Sim(a,b) = \alpha * F(\phi(a) \cap \phi(b)) - \beta * F(\phi(a) \setminus \phi(b)) \tag{3}$$

where $F()$ is a defined function of the feature sets. $\phi(concept)$ represents the feature sets of the concept. "$\setminus$" and "$\cap$" are operations to get overlapping and non-overlapping set, respectively. α and β are weights of each part. The main idea is that overlapping features increase semantic similarity and the non-overlapping features ones decrease it.

The limitation of this kind of approach is that they assume that each concept in the document has equal contribution to the semantic similarity. However, this is not the case. For instance, some frequently used concepts (low level nodes in the tree structure) should contribute less than those infrequently used concepts. In addition, using of feature sets completely ignores the influence of the organization structure of all concepts.

Information content-based approaches. These approaches consider the quantification of semantic information that concepts have in common [15, 16, 17, 18]. IC (Information Content) of a concept "a" is usually computed by $-\log(P(a))$, where $P(a)$ is the probability of occurrence of concept "a" in a given ontology. Semantic similarity of concept pairs "a" and "b" is computed through Formula (4):

$$Sim(a,b) = IC(LCS(a,b)) \tag{4}$$

where LCS (Least Common Subsume) is the most specific taxonomical ancestor common to both "a" and "b".

The core idea of these approaches is that the more concrete meaning of the common ancestor concept is, the higher semantic similarity between the concept pairs is. This implies that concept in the concept pairs is more concrete than their common ancestor concepts, because abstract concept lies in the upper layer of the directed graph. However, this kind of approach considers only the absolute distance between the LCS and compared concept pairs. It ignores the relative distance between the concept pairs.

Generally, concepts that are in a single ontology belong to the same domain. However, concepts that are used to describe Web Services are likely from different knowledge sources. This makes the existing approaches not applicable any more, because it is difficult to compute path distance, feature sets, and least common subsume in the case of multiple ontologies.

2.2. **Semantic similarity in multiple ontologies.** With the widespread of Semantic Web paradigm, numerous ontologies are constructed and available currently, for example, WordNet [26] (for general-purpose) and other specific domain ontologies. Researches are focusing on semantic similarity measuring of concept pairs by utilizing multiple ontologies to overcome the limitations of single ontology. Most of the proposed approaches are extension of works in a single ontology.

Extended edge-based approaches. The main idea of this kind of approach is connecting two ontologies by a bridge (the concept "*bridge*" is a virtual concept as left sub-figure in Figure 2), and then, using extended edge-based approach to measure semantic similarity [23]. The authors classified ontologies into primary ontology and secondary ontology. Firstly, the secondary ontology is connected to the primary ontology by joining the common concepts (the same concept in two ontologies as "a_2" and "b_1" in Figure 2) in two ontologies. They use Formula (5) to calculate the semantic distance between concepts "a" and "b".

$$Distance(a, b) = d_a + d_b - 1 \tag{5}$$

where $d_a = Distance(a, bridge)$ and $d_b = Distance(b, bridge)$ are measured based on Formula (1). $bridge$ is a virtual concept as right sub-figure in Figure 2.

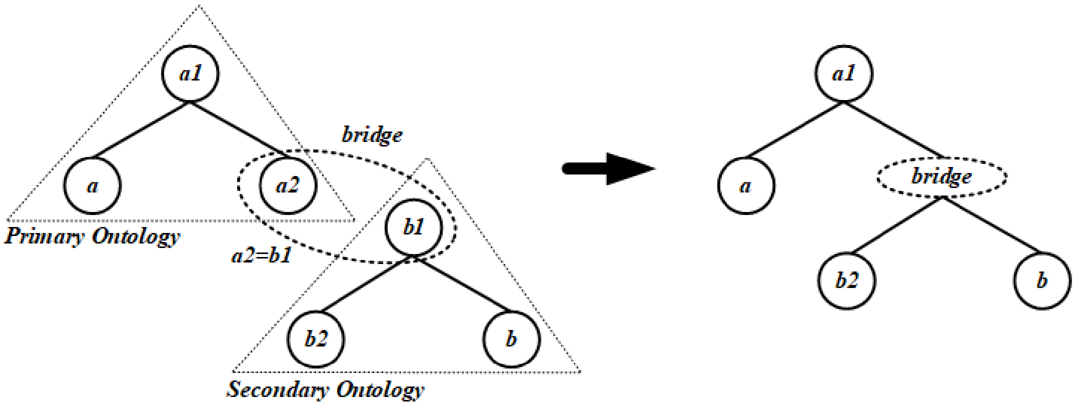


FIGURE 2. Connecting two ontologies with a bridge

This approach breaks the limitation of original edge-based approaches in which only semantic similarity between concept pairs in the same ontology can be measured. However, a primary ontology must be selected first in this approach. And, the authors assume that the primary ontology selected will always provide better result (that is not always the truth). There must be a pair of common concepts at least; nevertheless, the relationship of common concepts cannot be decided before performing semantic similarity measuring. In addition, this approach has the same limitation as original edge-based approach.

Extended feature-based approaches. The authors proposed a multiple ontologies semantic similarity measuring approach named X-Similarity that relies on matching synsets and term description sets in [13]. The term description sets are words extracted by parsing concept definitions. Two concepts are similar if their synsets or description sets, or the synsets of the concepts in their neighborhood (e.g., more specific and more general terms) are lexically similar. Semantic similarity between concept "a" and concept "b" is measured as Formula (6):

$$Sim(a,b) = \begin{cases} 1, & \text{if } \frac{|A \cap B|}{|A \cup B|} > 0 \\ \max\left\{\max\left\{\frac{|A_i \cap B_i|}{|A_i \cup B_i|}\right\}, \frac{|A \cap B|}{|A \cup B|}\right\}, & \text{if } \frac{|A \cap B|}{|A \cup B|} = 0 \end{cases} \tag{6}$$

where A and B denote synsets or concept description sets of concepts "a" and "b", $\frac{|A \cap B|}{|A \cup B|} > 0$ denotes the synsets similarity, $\max\left\{\frac{|A_i \cap B_i|}{|A_i \cup B_i|}\right\}$ means the description sets similarity, and "i" means the relationship type (e.g., IS-A and Part-Of). For instance, A_{IS-A} represents the direct child concept set of "a". $\frac{|A \cap B|}{|A \cup B|}$ and $\frac{|A_i \cap B_i|}{|A_i \cup B_i|}$ are computed according to Formula (3). The semantic similarity between concepts "a" and "b" is the larger value between synsets similarity and description similarity.

This kind of approach enhances the original feature-based approaches by utilizing the subsidiary information (i.e., synsets and description set) which considers the concept in the same ontology and the description of each concept. However, it does not overcome the limitation of the original feature-based approach.

Extended information content-based approaches. An extended information content-based approach is proposed to measure semantic similarity in [24]. The proposed approach relies on information theory that utilizes notion of mutual information. They estimate semantic similarity between concepts "c_1" and "c_2" in different ontologies as Formula (7).

$$iIC(MICA(c_1, c_2)) = \min\{iIC(cs_i), iIC(cs_j)\} \tag{7}$$

where $iIC(c)$ is intrinsic IC of concept "c" in an ontology modeled in [21, 27]. Here, most informative common ancestor of concepts "c_1" and "c_2" ($MICA(c_1, c_2)$) has the similar effect as least common subsume (LCS). cs_i and cs_j are subsumers of concepts "c_1" and "c_2" respectively.

The semantic similarity between concepts c_1 and c_2 is estimating the least value of all intrinsic IC (iIC) of the $MICA(c_1, c_2)$. This approach adopts a conservative method by accepting the least value of $iIC(MICA(c_1, c_2))$.

$$iIC(c) = -\log\left(\frac{\frac{|leaves(c)|}{|subs(c)|} + 1}{max\_leaves + 1}\right) \tag{8}$$

Formula (8) gives the methods to compute $iIC(c)$. Here, $|leaves(c)|$ is the number of directly or indirectly child nodes of "c", and $|subs(c)|$ is the number of directly or indirectly parent nodes of "c".

This kind of approach considers parent concepts and child concept of the compared concepts when estimating semantic similarity. Due to no bridge to connect two different ontologies, the author adopts a conservative method to estimate the semantic similarity. This may lead to the result of estimation is less than the actual value. In addition, they do not fully consider the structure features of concept, such as absolute depth, density, and the sibling nodes, in the tree structure. This approach also relies heavily on well-defined ontologies.

2.3. **Semantic similarity in user-defined knowledge source.** Most of the previous works neglect a practice that there are many user-defined knowledge sources (e.g., WSDL documents developed by programmers who have no knowledge of semantic or ontology).

This makes aforementioned methods hard to work or prone to bad results because user-defined knowledge may be very simple (e.g., no synsets, no glosses, and no concept description) and obscure semantic information (i.e., the semantic of concept cannot express without some related concepts directly).

In this paper, we take semantic annotation of user-defined WSDL document as the background. The main work of WSDL document semantic annotation is measuring semantic similarity between concept pairs from WSDL documents and ontology in a knowledge base (e.g., OWL files). There have been some previous works that concentrate on this issue; however, most of them pay attention to tools making rather than the measuring algorithm except for [27, 28, 29, 30].

Patil et al. proposed a framework for semi-automatically marking up Web Services description with ontologies called MWSAF in [27]. They use a combination of lexical-level and structural-level similarity measuring, where ontology concepts will be used to annotate WSDL document. MWSAF introduced the semantic annotation framework to annotate the input and output data in the WSDL document. The semantic similarity is measured by Formula (9):

$$MS = \frac{w_1 * ElemMatch + w_2 * SchemaMatch}{w_1 + w_2} \tag{9}$$

where *ElemMatch* and *SchemaMatch* represent lexical-level similarity and structural-level similarity of two concepts, respectively. *ElemMatch* adopts techniques of NGram and Abbreviation Expansion. *SchemaMatch* considered the similarity of sub-concepts and the ratio of matched sub-concepts. However, the organization structure of elements was not fully utilized both in WSDL and OWL document. This may decrease the accuracy of the semantic similarity measuring.

The authors proposed a lexical-based alignment semantic annotation approach in [28]. They generate synonyms of a concept according to WordNet (that provides separate definitions for each sense of the word). Then, a 2D matrix that holds the synonyms of the word for each sense in one dimension, and derivation hierarchies of the senses in other dimension was obtained by the synonyms. In the lexical-based alignment, they did matching over level-sense synsets by using name equality between all elements in the generated synonyms. A table, in which each cell is a tetrad containing name equality concept pairs and their levels, will be obtained. At last, the semantic similarity of synonyms is calculated by Formula (10):

$$md\left(c_{a_{sense_i}}, c_{b_{sense_j}}\right) = \left(\frac{2 * d_{nl}}{d_{sl^1} + d_{sl^2}}\right)^2 \tag{10}$$

where d_{nl} denotes the derivation order of common node, and d_{sl^1} and d_{sl^2} denote the derivation order of the first and second sense leaves, respectively.

In [29, 30], the authors proposed a semi-automatic WSDL Web Services description documents. Firstly, they classify WSDL services description (which is broken down into XSD data types, interfaces, operations and messages) to its corresponding domain. And then, the semantic similarity between a WSDL concept and the concepts of the selected domain ontology will be computed to identify which ontology concept to annotate the WSDL concept. The algorithm of semantic similarity measuring is not detailed in the paper.

The limitations of all the above approaches are that they do not fully utilize information of relevant concepts in both WSDL document and domain ontology file. For example, [27] does not consider the importance of each concept in both WSDL and ontology, and, the authors do not consider the sub-concepts of compared concept pairs in [28].

The proposed approach aims to address the above limitations of annotating WSDL documents, and improve the accuracy of semantic similarity measuring by fully utilizing the semantic information (includes lexical-level and structural-level). Especially, structural semantic information of a concept is represented by the concept itself and its internal features, depth, width, and density, in the tree structure.

3. **Our Solution.** Figure 3 illustrates the annotation framework of the proposed approach that contains three major steps. Step 1 represents the corresponding items in WSDL document and OWL file with an abstract tree structure respectively. Step 2 measures semantic similarity degree (abbreviates for SSD hereinafter) between nodes in the different tree structures. Specifically, the lexical-level similarity of each concept in WSDL document and each concept in ontology will be measured at first in this step. Then, we measure the structural-level similarity according to the measured lexical-level similarity. At last, semantic similarity is measured based on the results of lexical-level and structural-level similarity. In the last step, concepts in WSDL documents can be annotated with selected concepts in an ontology based on the results of semantic similarity measurement.

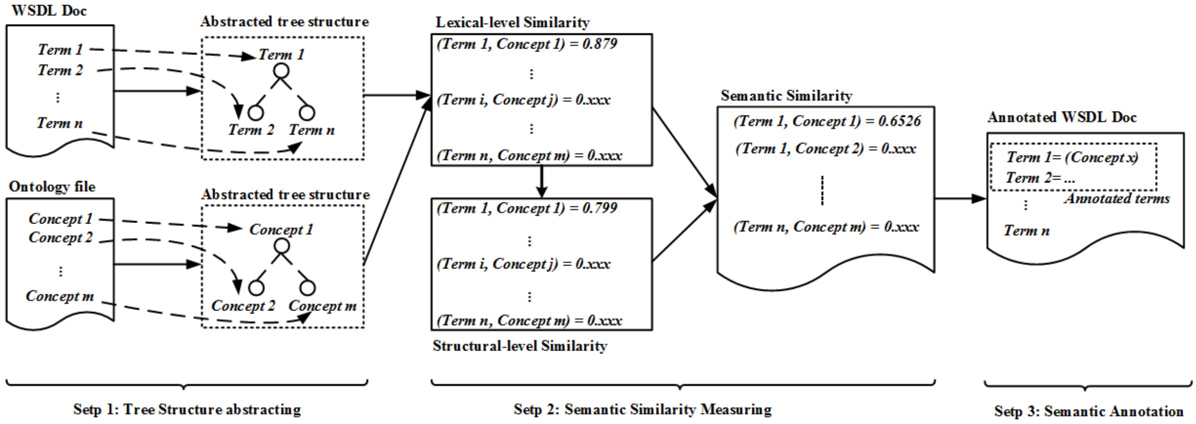

WSDL Doc
Term 1
Term 2
⋮
Term n
Abstracted tree structure
Term 1
Term 2
Term n
Ontology file
Concept 1
Concept 2
⋮
Concept m
Abstracted tree structure
Concept 1
Concept 2
Concept m
Lexical-level Similarity
(Term 1, Concept 1) = 0.879
⋮
(Term i, Concept j) = 0.xxx
⋮
(Term n, Concept m) = 0.xxx
(Term 1, Concept 1) = 0.799
⋮
(Term i, Concept j) = 0.xxx
⋮
(Term n, Concept m) = 0.xxx
Structural-level Similarity
Semantic Similarity
(Term 1, Concept 1) = 0.6526
(Term 1, Concept 2) = 0.xxx
(Term n, Concept m) = 0.xxx
Annotated WSDL Doc
Term 1= (Concept x)
Term 2= ...
Annotated terms
⋮
Term n
Setp 1: Tree Structure abstracting
Setp 2: Semantic Similarity Measuring
Setp 3: Semantic Annotation

Figure 3. The logic flow chart of the proposed approach

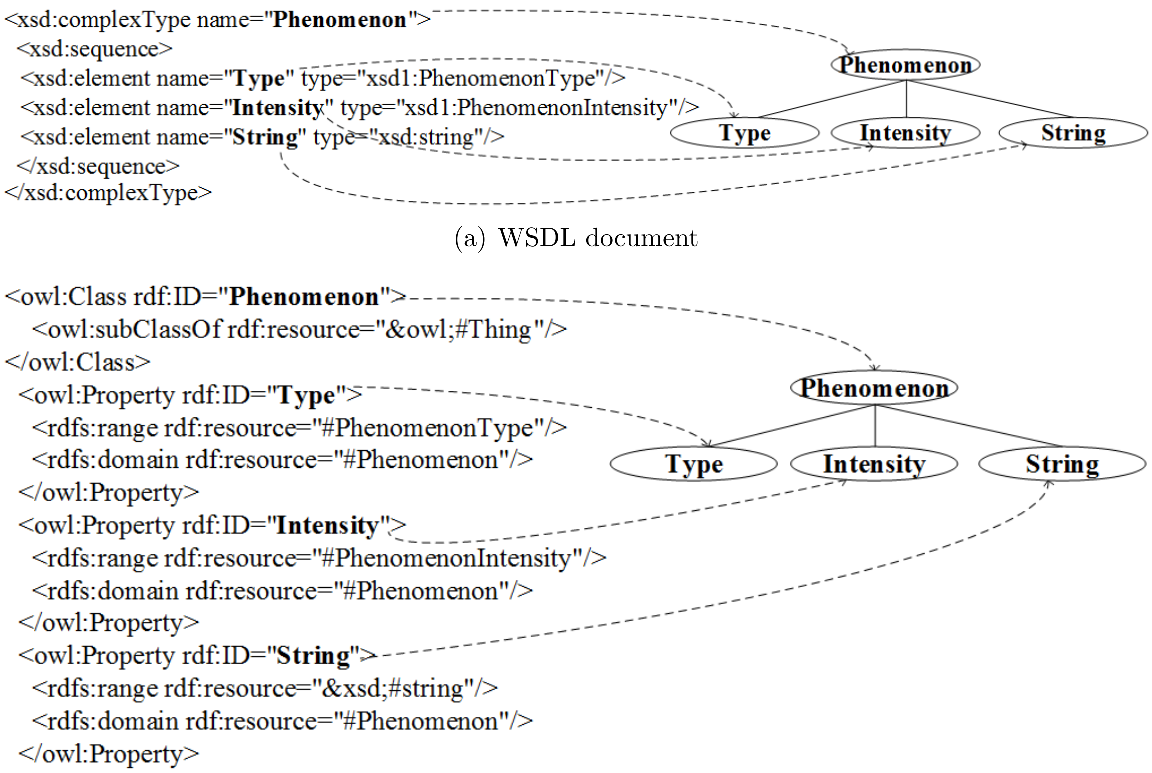
Table 1. Tree structure mapping rules of WSDL concepts

WSDL Document	**WSDL Tree (WT) Structure**
ComplexType	Node with ComplexType name
Element defined under ComplexType	Node with Element name
ComplexType defined under ComplexType	Node with name, such as Figure 4(a)
SimpleType	Node with SimpleType name
Values defined for simple types	Node with value as its name
Element	Node with Element name
Enumerated	Node with name
Relationship	Edge

3.1. **Tree structure representation of WSDL and OWL.** Since the different representations of WSDL concepts and OWL concepts, direct semantic similarity measuring between WSDL and OWL concepts is very difficult [31]. A good solution is to map both of them into a common expression (an abstract tree structure) like the approaches in [27, 32]. The first step of the proposed approach is mapping concepts in WSDL and

Table 2. Tree structure mapping rules of OWL concepts

Ontology File	**OWL Tree (OT) Structure**
Class	Node with Class name
Property with basic data type as range	Node with Property name
Property with Class as range	Node with name, such as Figure 4(b)
Instance	Node with Instance name
Class-subClass, Class-Property, and Class-Instance relationship	Edge



```
<xsd:complexType name="Phenomenon">
 <xsd:sequence>
  <xsd:element name="Type" type="xsd1:PhenomenonType"/>
  <xsd:element name="Intensity" type="xsd1:PhenomenonIntensity"/>
  <xsd:element name="String" type="xsd:string"/>
 </xsd:sequence>
</xsd:complexType>
```

(a) WSDL document

```
<owl:Class rdf:ID="Phenomenon">
  <owl:subClassOf rdf:resource="&owl;#Thing"/>
</owl:Class>
 <owl:Property rdf:ID="Type">
  <rdfs:range rdf:resource="#PhenomenonType"/>
  <rdfs:domain rdf:resource="#Phenomenon"/>
 </owl:Property>
 <owl:Property rdf:ID="Intensity">
  <rdfs:range rdf:resource="#PhenomenonIntensity"/>
  <rdfs:domain rdf:resource="#Phenomenon"/>
 </owl:Property>
 <owl:Property rdf:ID="String">
  <rdfs:range rdf:resource="&xsd;#string"/>
  <rdfs:domain rdf:resource="#Phenomenon"/>
 </owl:Property>
```

(b) OWL file

Figure 4. A tree structure abstraction example

OWL documents to intermediate tree structures representation according to the revised mapping rules based on [27, 32].

Table 1 and Table 2 show the mapping rules of WSDL and OWL documents, where WT and OT are abbreviations of WSDL tree structure and OWL tree structure, respectively. Unlike [27, 32], we do not take edge's name into account, because the contribution of the edge's information to semantic similarity degree is temporarily not considered. An example of tree structure extraction is illustrated in Figure 4.

3.2. **Semantic similarity measuring.** After the tree structure is obtained, semantic similarity measuring will be performed between nodes in WT and OT. Each node in WT will do semantic similarity measuring with nodes in OT. The result of the semantic similarity measuring, i.e., SSD, is a value range of $[0, 1]$. For the node pairs of WT and OT, a higher value of semantic similarity degree means more semantic similarity between two concepts. The aim of this work is finding the most similar concept pairs that come from WT and OT respectively.

SSD between node pairs (W_i, O_j) will be denoted as $SSD(W_i, O_j)$, W_i and O_j are the name of node in WT and OT respectively, and $W_i \in W = \{W_1, W_2, \ldots, W_n\}$ and $O_j \in O = \{O_1, O_2, \ldots, O_m\}$.

In the proposed approach, semantic similarity degree of the term pairs $(SSD(W_i, O_j))$ consists of two parts: lexical-level similarity and structural-level similarity. Lexical-level similarity $(S_l(W_i, O_j))$ indicates the linguistic similarity between the two nodes, and structural-level similarity $(S_s(W_i, O_j))$ means the structural similarity. Specifically, $SSD(W_i, O_j)$ can be measured by Formula (11):

$$SSD(W_i, O_j) = w_l * S_l(W_i, O_j) + (1 - w_l) * S_s(W_i, O_j) \tag{11}$$

where $w_l \in [0, 1]$ denotes the contribution of the lexical-level similarity in $S_s(W_i, O_j)$ to $SSD(W_i, O_j)$.

Table 3 illustrates details about the experience value of w_l. It is a dynamic value because the contribution of $S_l(W_i, O_j)$ and $S_s(W_i, O_j)$ interacts with each other. Principles of setting the value of w_l are as follows:

(1) w_l is dynamic that changes with the ratio of S_l and S_s. Especially, the larger $\frac{S_l}{S_s}$ is, the smaller w_l is;

(2) Structural-level contributes more to the final semantic similarity when $S_l = S_s$.

Table 3. Dynamic value setting of weight w_l

w_l S_l S_s	**0**	**(0, 0.4]**	**(0.4, 0.6]**	**(0.6, 0.8]**	**(0.8, 1]**
0	0.4	0.5	0.6	0.8	1
(0, 0.3)	0.3	0.4	0.5	0.7	0.9
[0.3, 0.6)	0.2	0.3	0.4	0.5	0.7
[0.6, 0.9)	0.1	0.2	0.3	0.4	0.5
[0.9, 1]	0	0.1	0.2	0.3	0.4

Note that, the division of the interval of S_l and S_s in Table 3 can be changed case by case. This depends on the granularity that the algorithm wants to be. The stride of each interval can be small if the semantic similarity is sensitive to the threshold that is a value range of $[0, 1]$. For example, concept "*a*" will be annotated with concept "*b*" if $SSD(a, b) = 0.92$ when the threshold is set to 0.91.

3.2.1. *Lexical-level similarity measuring.* The lexical-level similarity is the measurement of linguistic similarity between WSDL and ontology concept. There are many approaches to measure the linguistic similarity of two words, such as NGram [36], synonym matching [37], Levenstein Distance [33, 34], and Abbreviation Expansion [38].

We use Levenstein Distance and Abbreviation Expansion to measure lexical-level similarity based on the assumption that the string used for naming concepts in WSDL or concepts in OWL ontologies is single word or words connected with special character, i.e., space, capital letter etc. The lexical-level similarity $S_l(W_i, O_j)$ is calculated as Formula (12):

$$S_l(W_i, O_j) = Max\,\{LD_{sim}(W_i, O_j), AE_{sim}(W_i, O_j)\} \tag{12}$$

where

$$LD_{sim}(W_i, O_j) = 1 - \frac{ld(W_i, O_j)}{MaxLength(W_i, O_j)} \tag{13}$$

denotes the lexical-level similarity from Levenstein Distance. $ld(W_i, O_j)$ [34] denotes the Levenstein Distance between W_i and O_j, and $MaxLength(W_i, O_j)$ means the largest string length of the two concepts.

$$AE_{sim}(W_i, O_j) = \begin{cases} 0, & \text{if no abbreviation between } W_i \text{ and } O_j; \\ 1, & \text{else} \end{cases} \tag{14}$$

For example, let "*PC*" be abbreviation of "*PersonalComputer*". Therefore, we can get the results that $AE_{sim}(PersonalComputer, PC) = 1$ and $AE_{sim}(PersonalPlane, PC) = 0$.

However, there is a limitation if only the lexical-level similarity is considered. Taking concept pair *(string, strong)* for example, the value of $LD_{sim}(string, strong) = 5/6$ that cannot mean the semantic similarity between "string" and "strong" exactly while they have completely different meanings. Furthermore, even if the value of $LD_{sim}(W_i, O_j) = 1$, the meaning of the two concepts may be different. Taking $LD_{sim}(chair, chair) = 1$ for example, the semantic of the two "chair" may be different. One resembles an ordinary seat for a person while another means an officer/leader of an organization.

Therefore, we cannot consider only the lexicon information of a concept when measuring semantic similarity. We can utilize the information accompanied by the concept in the same document. That is, we should take all the relevant nodes in the abstracted tree structure into account when measuring the semantic similarity degree.

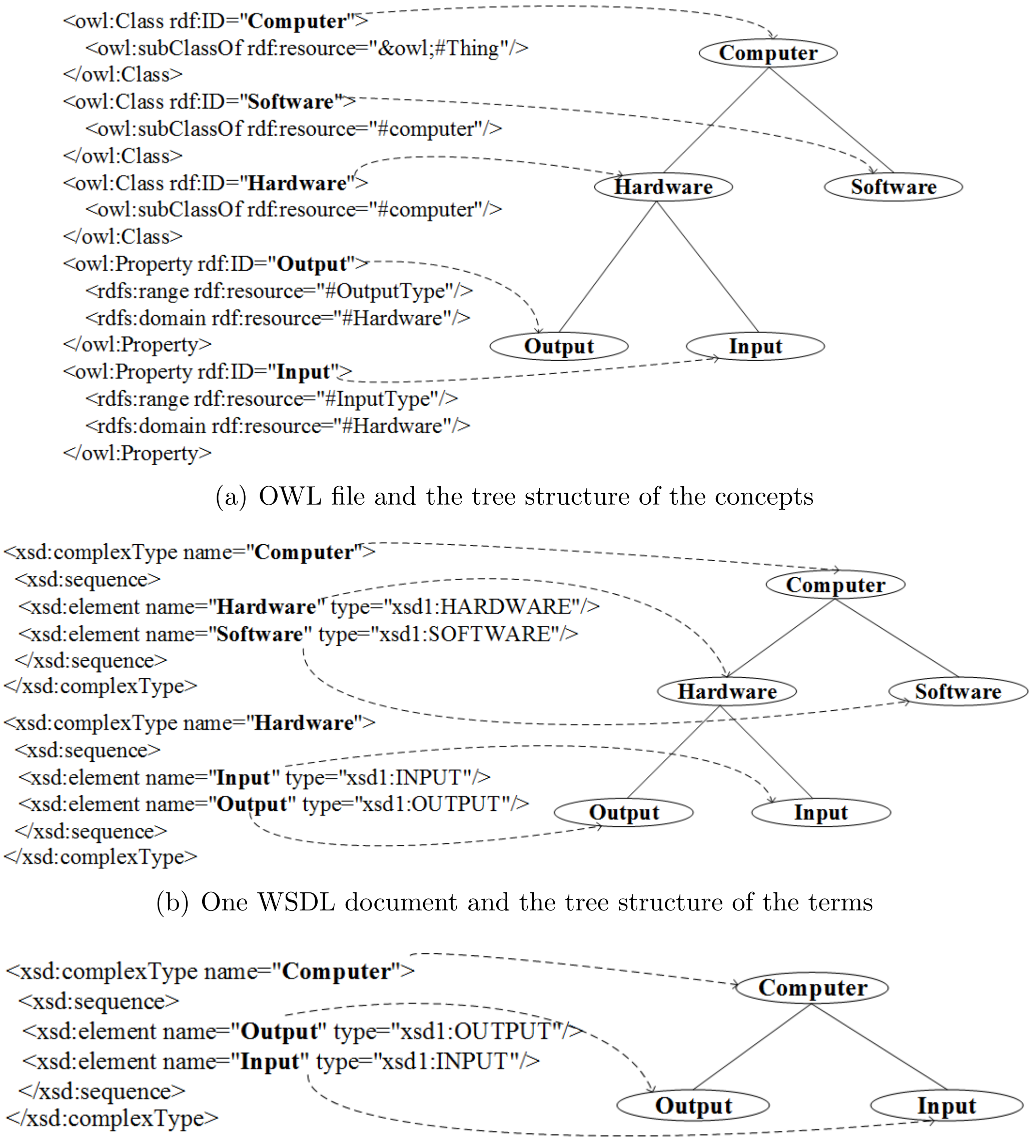
3.2.2. *Structural-level similarity measuring.* The structural-level similarity is a measurement of structural similarity between two nodes while lexical-level similarity cannot completely present the semantic similarity between two concepts. We should take use of not only the string of the node's name but information of its relevant nodes[1]. Furthermore, the location, relationships, and contribution of the relevant nodes will be utilized when conducting structural-level similarity measuring.

Structure analysis of tree structure: Generally, domain ontology is built by domain experts; however, WSDL documents are produced by different organizations or persons with different perspectives. There is probability to cause different understanding of the same object between the WSDL document designers and domain experts. The ambiguity of personal understanding often cannot be literally displayed. We need additional information to determine the semantic of an object in a description document. In this paper, we use relevant nodes and their relationships as the additional information. The relationship of concepts in a description document is represented as an abstract tree structure. We map each concept to a node in the tree structure.

The mapping of a segment in OWL file is illustrated in Figure 5(a), and Figure 5(b) and Figure 5(c) give two different WSDL document segments and the corresponding tree structure mappings. We can find that mapping of WSDL document in Figure 5(b) is the same as the mapping of OWL file when Figure 5(c) has little difference with Figure 5(a) and Figure 5(b).

It should be noted that the node "*Computer*" in Figure 5(b) has the same tree structure with the node "*Computer*" in Figure 5(a). And, the SSD may be different between $SSD\left(Computer^A, Computer^B\right)$ and $SSD\left(Computer^A, Computer^C\right)$ when we consider the relevant concepts, such as "*Software*", "*Hardware*", "*Input*", and "*Output*". Hence,

[1] In principle, nodes that have paths to O_j called relevant nodes of O_j. In this paper, only nodes in the sub-layers of O_j are considered. Nodes in the upper-layers do not belong to the relevant nodes set. For example, "*Computer*", "*Software*", "*Input*", and "*Output*" have paths to "*Hardware*" in Figure 5(a). However, only "*Input*" and "*Output*" are regarded as relevant nodes of "*Hardware*" in this paper.



```
<owl:Class rdf:ID="Computer">
  <owl:subClassOf rdf:resource="&owl;#Thing"/>
</owl:Class>
<owl:Class rdf:ID="Software">
  <owl:subClassOf rdf:resource="#computer"/>
</owl:Class>
<owl:Class rdf:ID="Hardware">
  <owl:subClassOf rdf:resource="#computer"/>
</owl:Class>
<owl:Property rdf:ID="Output">
  <rdfs:range rdf:resource="#OutputType"/>
  <rdfs:domain rdf:resource="#Hardware"/>
</owl:Property>
<owl:Property rdf:ID="Input">
  <rdfs:range rdf:resource="#InputType"/>
  <rdfs:domain rdf:resource="#Hardware"/>
</owl:Property>
```

(a) OWL file and the tree structure of the concepts

```
<xsd:complexType name="Computer">
 <xsd:sequence>
  <xsd:element name="Hardware" type="xsd1:HARDWARE"/>
  <xsd:element name="Software" type="xsd1:SOFTWARE"/>
 </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Hardware">
 <xsd:sequence>
  <xsd:element name="Input" type="xsd1:INPUT"/>
  <xsd:element name="Output" type="xsd1:OUTPUT"/>
 </xsd:sequence>
</xsd:complexType>
```

(b) One WSDL document and the tree structure of the terms

```
<xsd:complexType name="Computer">
 <xsd:sequence>
  <xsd:element name="Output" type="xsd1:OUTPUT"/>
  <xsd:element name="Input" type="xsd1:INPUT"/>
 </xsd:sequence>
</xsd:complexType>
```

(c) The other WSDL document and the tree structure of the terms

FIGURE 5. Tree structure with OWL file and WSDL document fragment

the difference of the tree structure should be reflected in semantic similarity degree, even if they have the same value of lexical-level similarity.

To accurately measure a semantic similarity degree between concept pairs, we will take full advantage of the inherent features of nodes in the tree structure. Especially, we will utilize the relationship of depth, width, and density of nodes in the tree structure when measuring the structural-level similarity.

Structural-level similarity matching: Let $S_s(W_i, O_j)$ denote the structural-level similarity between two terms W_i and O_j in WT and OT respectively. $S_s(W_i, O_j)$ is calculated by Formula (15):

$$S_s(W_i, O_j) = \frac{F(W_i, O_j)}{NumOf\left(S_l\left(W_k', O_l'\right)\right)} \tag{15}$$

where $W_k' \in W' = \left\{W_i' | W_i' \text{ is the relevant nodes of } W_i\right\}$ and $W' \subset W$ is the true subset of W in which W_i is not included, and O_j is the same; $NumOf\left(S_l\left(w_k', O_l'\right)\right)$ is the number of relevant nodes of W_i that has a lexical-level similarity larger than a threshold. The threshold changes case by case. A large value reduces the time that is used for computation. A small value of the threshold provides more comprehensive coverage and candidate results with heavy computation.

$$F\left(W',O'\right) = Max\left\{\sum_{i=1}^{NumOf\left(S_l\left(W'_k,O'_l\right)>0\right)} w\left(W'_k,O'_l\right) * S_l\left(W'_k,O'_l\right)\right\} \tag{16}$$

$F\left(W',O'\right)$ is a function to select the maximum value of summation of lexical-level similarity $S_l\left(W'_k,O'_l\right)$ with weight $w\left(W'_k,O'_l\right)$ (a weight to reflect the influence of the relevant nodes organization structure in the corresponding tree structure). $w\left(W'_k,O'_l\right)$ is constituted of three parts as the following Formula (17).

$$w\left(W'_k,O'_l\right) = w_d\left(W'_k,O'_l\right) * w_w\left(W'_k,O'_l\right) * w_\rho\left(W'_k,O'_l\right) \tag{17}$$

where $w_d\left(W'_k,O'_l\right) \in [0,1]$, $w_w\left(W'_k,O'_l\right) \in [0,1]$ and $w_\rho\left(W'_k,O'_l\right) \in [0,1]$ are the weight values of node's depth, width, and density, respectively.

w_d: inherent feature of *depth* $\left(w_d\left(W'_k,O'_l\right)\right)$ is a weight that reveals the contribution of node's depth in the tree structure to the structural-level similarity.

Definition 3.1. *Depth, the level difference between node and ROOT node (with level = 0 and name is "Thing") in a tree structure, is denoted as* $Dep(x) = LevelOf(x) - LevelOf(ROOT)$.

For example, $Dep(Computer) = LevelOf(Computer) - LevelOf(ROOT) = n$ of node "*Computer*" in Figure 6. Rada et al. believed that the longer the path is, the semantically farther the concepts are [8], and Li et al. believed that concepts at upper layers of the hierarchy have more general semantics, when concepts at lower layers have more concrete semantics [10]. We believe that the node's depth will influence the semantic similarity degree between concept pairs.

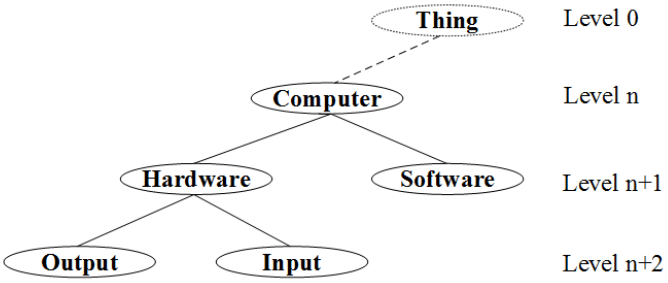


FIGURE 6. Depth of nodes in the tree structure (dashed line represents the omitted part of the tree)

We adjust the value of $w_d\left(W'_k,O'_l\right) \in [0,1]$ according to the depth difference between W_i and O_j by the following principles:

(1) if $Dep\left(W'_k\right) = Dep\left(O'_l\right)$, $w_d\left(W'_k,O'_l\right) = 1$;
(2) if $Dep\left(W'_k\right) \neq Dep\left(O'_l\right)$, the larger $\Delta Dep = \left|Dep\left(W'_k\right) - Dep\left(O'_l\right)\right|$ is, the smaller $w_d\left(W'_k,O'_l\right)$ is;
(3) if $Dep\left(W'_k\right) \neq Dep\left(O'_l\right)$, the larger $\Sigma Dep = Dep\left(W'_k\right) + Dep\left(O'_l\right)$ is, the smaller $w_d\left(W'_k,O'_l\right)$ is.

The core principle is that the larger depth difference is, the smaller $w_d\big(W_k', O_l'\big)$ is. Based on the above analysis, we proposed Formula (18) as:

$$w_d\big(W_k', O_l'\big) = f_d\left(f_\Delta(\Delta Dep), f_\Sigma(\Sigma Dep)\right) \tag{18}$$

where $\Sigma Dep = Dep\big(W_k'\big) + Dep\big(O_l'\big)$ and $f_\Sigma(\Sigma Dep)$ indicates the principle (3) while $f_\Delta(\Delta Dep)$ indicates principle (1) and principle (2).

Taking the above considerations into account, we set $f_\Sigma(\Sigma Dep)$ and $f_\Delta(\Delta Dep)$ to be monotonically decreasing and monotonically increasing as Formula (19) and Formula (20), respectively:

$$f_\Delta(\Delta Dep) = e^{-\alpha * |\Delta Dep|} \tag{19}$$

$$f_\Sigma(\Sigma Dep) = 1 - e^{-\beta * \Sigma Dep} \tag{20}$$

where $w_d\big(W_k', O_l'\big)$ is considered to be governed by the ΔDep and ΣDep, as Formula (21):

$$w_d\big(W_k', O_l'\big) = \gamma * e^{-\alpha * |\Delta Dep|} * \big(1 - e^{-\beta * \Sigma Dep}\big) \tag{21}$$

where γ is an adjustment factor to control the value of $w_d\big(W_k', O_l'\big)$. Experimental values are $\gamma = 1$, $\alpha = 0.3$, and $\beta = 1$.

$\boldsymbol{w_w}$**: inherent feature of *width*** $\big(w_w\big(W_k', O_l'\big)\big)$ is a weight that is associated with the node's width.

Definition 3.2. *Width, the number of one node's sibling nodes in the tree structure, is denoted as* $Wid(x)$.

Based on the idea of feature-based approaches [12, 25], the common features increase the semantic similarity when non-common features decrease it. We give the following principles of $w_w\big(W_k', O_l'\big)$.

(1) $w_w\big(W_k', O_l'\big) = 1$, if the sibling nodes are the same;
(2) $w_w\big(W_k', O_l'\big) < 1$, if there exist different sibling nodes;
(3) The more of same sibling nodes a node has or the larger lexical-level similarity between corresponding nodes is, the larger $w_w\big(W_k', O_l'\big)$ is.

It is noted that the value of $w_w\big(W_k', O_l'\big)$ mainly depends on the sibling nodes of W_k' and O_l'.

The measurement of $w_w\big(W_k', O_l'\big)$ is given as Formula (22):

$$w_w\left(W_k', O_l'\right) = \frac{Max\left\{\sum_{i=1}^{NumOf\left(S_l\left(sib_f^{W_k'}, sib_g^{O_l'}\right) > 0\right)} S_l\left(sib_f^{W_k'}, sib_g^{O_l'}\right)\right\}}{(1+\alpha) * NumOf\left(S_l\left(sib_f^{W_k'}, sib_g^{O_l'}\right) > 0\right) + \alpha * |m - n|} \tag{22}$$

where α is set to be an experimental value as 0.5, $sib^{W_k'} = \left\{sib_m^{W_k'}\right.$ is the sibling node of $\left. sib^{W_k'}\right\}$ and $sib^{O_l'}$ is similar to $sib^{W_k'}$. m and n are the sizes of $sib^{W_k'}$ and $sib^{O_l'}$, respectively. $NumOf\left(S_l\left(sib_f^{W_k'}, sib_g^{O_l'}\right) > 0\right)$ is the number of sibling node pairs that have a lexical-level similarity larger than 0.

$\boldsymbol{w_\rho}$**: inherent feature of *density*** $\big(w_\rho\big(W_k', O_l'\big)\big)$ is a weight that is associated with the node's density.

Definition 3.3. *Density, appearance frequency of one node in the node set of a tree structure, is denoted as* $\rho(x)$.

Following the standard argumentation of information theory, the information content of a concept x can be quantified as negative log likelihood, $-\log \rho(x)$. Intuitively, a concept has less information content if it has a higher $\rho(x)$.

Researchers found that the semantic of a concept was decided by the attributes belonging to it, and the higher an appearance frequency of one attribute is, the less contribution to the semantic is. In addition, the more common information of two words' attributes is, the more similarity the two words are [15, 35]. We believe that the semantic similarity degree between two nodes in the corresponding tree structures is impacted by the density of their relevant nodes. Therefore, we define the following principles when measuring $w_\rho(W_k^{'}, O_l^{'})$ as:

(1) $w_\rho(W_k^{'}, O_l^{'}) = 1$, if the densities of two relevant nodes are the same;
(2) $w_\rho(W_k^{'}, O_l^{'}) < 1$, if the densities of two relevant nodes are not the same;
(3) the higher $\rho(W_k^{'})$ or $\rho(O_l^{'})$ is, the less information of $W_k^{'}$ and $O_l^{'}$ is.

Based on the principles, $w_\rho(W_k^{'}, O_l^{'})$ is calculated by Formula (23):

$$w_\rho\left(W_k^{'}, O_l^{'}\right) = \begin{cases} 1, & \text{if } \rho(W_k^{'}) = \rho(O_l^{'}); \\ f_\rho\left(\rho(W_k^{'}), \rho(O_l^{'})\right), & \text{if } \rho(W_k^{'}) \neq \rho(O_l^{'}) \end{cases} \tag{23}$$

and

$$f_\rho\left(\rho(W_k^{'}), \rho(O_l^{'})\right) = \log_{\frac{1}{N_w}}^{\rho\left(W_k^{'}\right)} * \log_{\frac{1}{N_o}}^{\rho\left(O_l^{'}\right)} \tag{24}$$

where N_w is the total number of the sub-nodes in WT and N_o is the total number of the sub-nodes in OT, and $\rho(W_k^{'})$ and $\rho(O_l^{'})$ are the densities of $W_k^{'}$ and $O_l^{'}$, respectively.

3.3. Formal description of the proposed approach.

The formal description of structural-level similarity calculating of concept pair $\left(W_i^{'},\ O_j^{'}\right)$ is illustrated in Algorithm 1.

In Algorithm 1, the function call $FindMaxOf(S_s[i][j])$ is used to find the maximum value of $F(W^{'}, O^{'})$ in $S_s[i][j]$. $F(W^{'}, O^{'})$ is the sum of $S_s[i][j].S_s(W_i^{'}, O_j^{'})$ with conditions as follows:

(1) only one element $S_s[i][j].S_s(W_i^{'}, O_j^{'})$ of each row in matrix $S_s[i][j]$ is selected;
(2) the times of $W_i^{'}$ and $O_j^{'}$ appearance in the selected set of $S_s[i][j]$ is no more than the times they appear in $W^{'}$ and $O^{'}$.

4. Experimental Evaluation.

In this experiment, we perform comparisons from two aspects: longitudinal comparison and horizontal comparison. We illustrate longitudinal comparison to validate that node's internal features of the tree structure indeed influence the semantic similarity in Subsection 4.2.1, and give the experimental results of comparison in aspect of semantic similarity degree among the proposed approach, previous approach and artificial scoring in Subsection 4.2.2.

4.1. Setting up.

Table 4 lists all the WSDL documents and OWL files used in our experiments[2]. SSD between "*parameter1*" and "*parameter2*" is represented as $SSD\,(parameter1, parameter2)$ where "*parameter1*" is a WSDL concept and "*parameter2*" is an OWL concept.

To simplify the representation in the following experiments, we will use the representation at the right part of the equation in Table 4 as a simplified representation of

[2] Most of the WSDL documents and the OWL file come from MWSAF project at *http://lsdis.cs.uga.edu/projects/meteors/downloads/*. "*WT1.wsdl*" and "*WT2.wsdl*" documents are modified by adding and deleting 1 element from "*Global-Weather.wsdl*", respectively, to change the structure of tree structure (actually, depth, width, and density of node in the structure will be changed).

Algorithm 1 Algorithm of $S_s(W_i, O_j)$ Measurement

```
Require: $W' = \left\{W'_1, W'_2, \ldots, W'_p\right\}$, $O' = \left\{O'_1, O'_2, \ldots, O'_q\right\}$, $S_s[p][q]$ /*p and q are the size of $W'$ and $O'$ respectively, and each term in the matrix $S_s$ is a structure type with 3 member variables $W_{name}$, $O_{name}$, and $S_s(W_i, O_j)$. $W'_i$ and $O'_j$ are both quaternary tuple with member variables (i.e., name, depth, width, and density)*/
Ensure: $S_s(W_i, O_j)$ /*the result of structural similarity between $W_i$ and $O_j$*/
for $i = 1 \rightarrow p$ do /*for each concept in $W'$*/
    for $j = 1 \rightarrow q$ do/*for each concept in $O'$*/
        $w_d\left(W'_i, O'_l\right) \leftarrow \gamma * e^{-\alpha*\left|Dep\left(W'_i\right)-Dep\left(O'_l\right)\right|} * \left(1 - e^{-\beta*\left(Dep\left(W'_i\right)+Dep\left(O'_j\right)\right)}\right)$; /*weight value of depth*/
        $w_w\left(W'_i, O'_l\right) \leftarrow \frac{Max\left\{\sum_{i=1}^{NumOf\left(S_l\left(Sib_f^{W'_k}, Sib_g^{O'_l}\right)>0\right)} S_l\left(Sib_f^{W'_k}, Sib_g^{O'_l}\right)\right\}}{(1+\alpha)*NumOf\left(S_l\left(Sib_f^{W'_k}, Sib_g^{O'_l}\right)>0\right)+\alpha*|m-n|}$;
        if $\rho\left(W'_k\right) == \rho\left(O'_l\right)$ then
            $w_\rho\left(W'_i, O'_l\right) \leftarrow 1$;
        else
            $w_\rho\left(W'_i, O'_l\right) \leftarrow \log_{\frac{1}{N_w}}^{\rho\left(W'_i\right)} * \log_{\frac{1}{N_o}}^{\rho\left(O'_j\right)}$;
        end if
        $w\left(W'_i, O'_j\right) \leftarrow w_d\left(W'_i, O'_j\right) * w_w\left(W'_i, O'_j\right) * w_\rho\left(W'_i, O'_j\right)$;
        $S_s[i][j].W_{name} \leftarrow W'_i$;
        $S_s[i][j].O_{name} \leftarrow O'_j$;
        $S_s[i][j].S_s\left(W'_i, O'_j\right) \leftarrow S_l\left(W'_i, O'_j\right) * w\left(W'_i, O'_j\right)$; /*structural-level similarity for concept pairs $\left(W'_i, O'_j\right)$*/
    end for
end for
$F\left(W'_i, O'_j\right) \leftarrow FindMaxOf(S_s[i][j])$; /*select the maximum value of $S_s[i][j]$ from all the measured values*/
$S_s\left(W'_i, O'_j\right) \leftarrow \frac{2*F\left(W'_i, O'_j\right)}{p+q}$; /*the final structural-level similarity of concept pair $\left(W'_i, O'_j\right)$*/
```

Table 4. Tree structure mapping rules of OWL

SSD(WSDL,OWL) **WSDL** **OWL**	**WeatherConcept.*owl***
WT1.*wsdl*	SSD(WeatherReport,WeatherReport)=SSD(WT1)
WT2.*wsdl*	SSD(WeatherReport,WeatherReport)=SSD(WT2)
GlobalWeather.*wsdl*	SSD(WeatherReport,WeatherReport)=SSD(GW)
AirportWeather.*wsdl*	SSD(WeatherSummary,WeatherReport)=SSD(AW)
WeatherFetcher.*wsdl*	SSD(Weather,WeatherReport)=SSD(WF)
FastWeather.*wsdl*	SSD(Weather,WeatherReport)=SSD(FW)
UnisysWeather.*wsdl*	SSD(GetWeatherResult,WeatherReport)=SSD(UW)

$SSD(parameter1, parameter2)$. For example, $SSD(WT1)$ represents the semantic similarity degree between the concept *"WeatherReport"* in *WT1.wsdl* document and the concept *"WeatherReport"* in *WeatherReport.owl*.

4.2. **Experimental results.** Longitudinal comparison and horizontal comparison will be conducted in this subsection. The purpose of the longitudinal comparison is to examine the impact of the internal features in structural-level similarity, and horizontal comparison will be conducted between the proposed approach and MWSAF in the aspect of semantic similarity. Additionally, we give some results of semantic similarity degree of artificial scoring as a reference in Appendix B.

4.2.1. *Longitudinal comparison.* Figure 7, Figure 8 and Figure 9 present the impact of depth, width, and density, separately. Labels in x-axis represent compared concept pairs

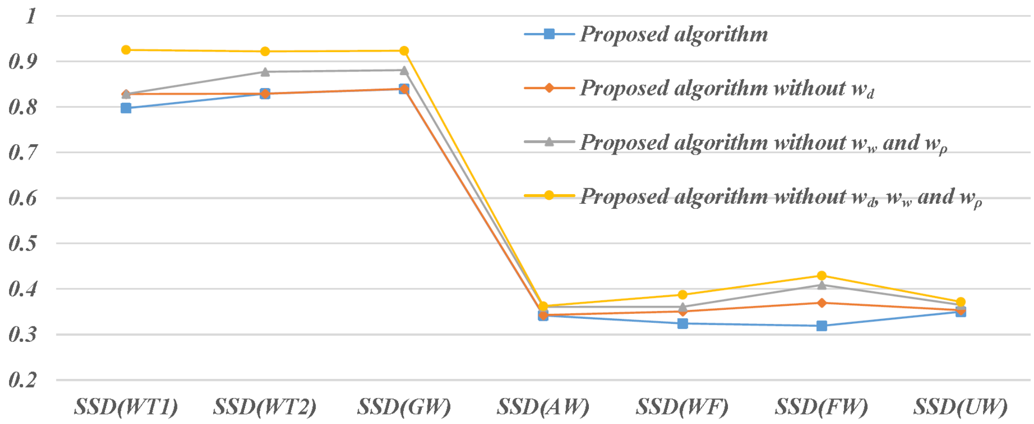


Figure 7. Influence of the internal feature *depth*

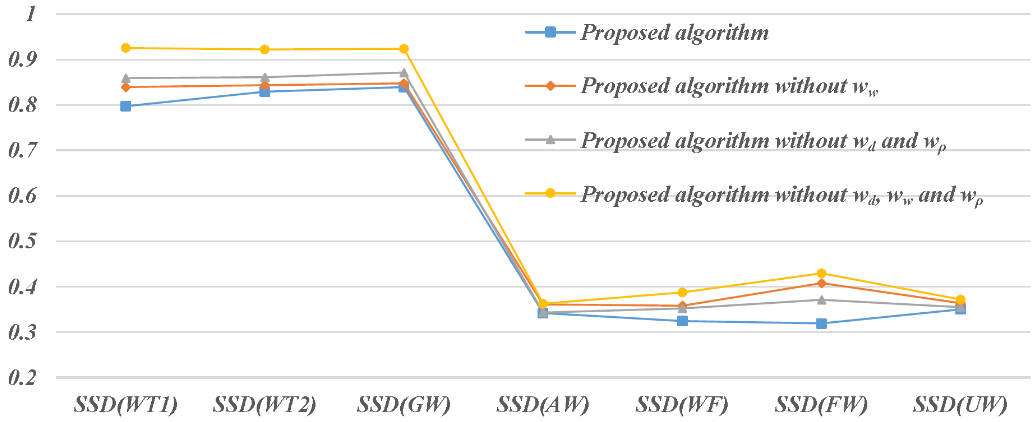


Figure 8. Influence of the internal feature *width*

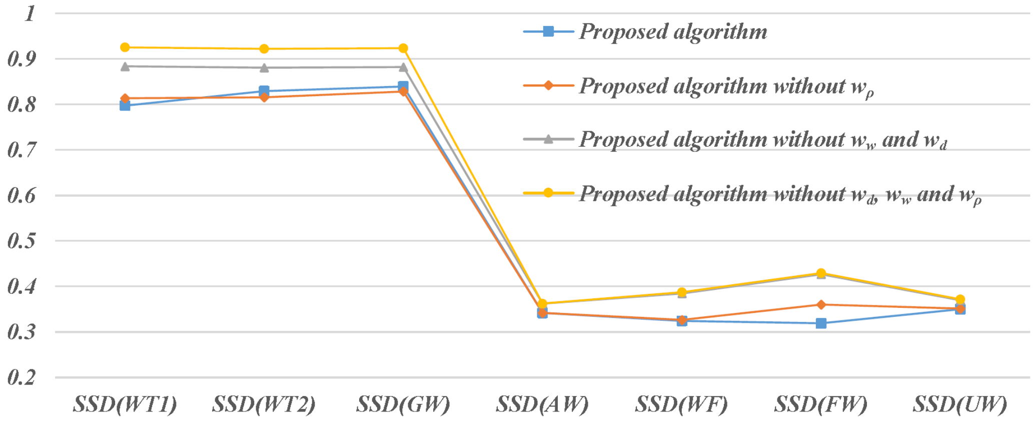


Figure 9. Influence of the internal feature *density*

in Table 4, and the y-axis is the value of corresponding structural-level similarity of the concept pairs.

From Figure 7, we can find node's depth indeed impacts the structural-level similarity. Structural-level similarity drops when considering the internal feature of node's depth. Because the weight value of depth is $w_d = 1$ if the internal feature "depth" is not considered, the proposed approach without considering the three kinds of internal features will get the highest structural-level similarity degree when the proposed approach, which considers all of the internal features, is the lowest one. Similar to Figure 7, Figure 8 and Figure 9 demonstrate the influence of width and density.

Combining Figure 7, Figure 8 and Figure 9, it validates that all the three kinds of internal features of a node in the tree structure impact the structural-level similarity. More specifically, they impact the structural-level similarity thereby impacting the semantic similarity degree between node pairs.

4.2.2. *Horizontal comparison.* This section presents a comparative study between approach in [27] and the proposed approach in the aspect of semantic similarity. Initially, we want to compare the approach proposed in [27] with all other approaches. However, unavailability of technical details and difficultly to exploit the associated tools with all these approaches prevent a complete study. Thus, the comparison is limited to study MWSAF approach that is a framework for semi-automatically annotating WSDL document of Web Services with domain ontologies.

To intuitively display the difference, we give a comparison among MWSAF and ArtificialScoring (detail in Appendix B), and the proposed approach. Due to lack of related benchmarks, 9 evaluators major of computer science are engaged to artificially assess values of the semantic similarity degree between concept pairs in Table 4. Specifically, 7 evaluators were graduate students in which 1 has experience with WSDL and 1 student with experience of SAWSDL. The rest of the evaluators were 2 Ph.D. students. It is hard to find semantic web experts evaluators at this stage of development but on the other hand, this less-skilled set of users can demonstrate how reality and usable the approach is.

From Figure 10, we can find that the results of all the three approaches (MWSAF, ArtificialScoring, and the proposed approach) change in a similar trend in aspect of semantic similarity degree. Semantic similarity degree of $SSD(WT1)$, $SSD(WT2)$, and $SSD(GW)$ in MWSAF has obvious difference. Table 6 in the Appendix B illustrates the results of artificial scoring, MWSAF, and the proposed approach, and we can find that the proposed approach has less deviation with the artificial scoring of semantic similarity of $SSD(WT1)$, $SSD(WT2)$, and $SSD(GW)$. Because, MWSAF considers only the direct child concepts in the file when the tree structures in $WT1.wsdl$ and $WT2.wsdl$ are obviously different with that in the original WSDL document. The fact is that MWSAF

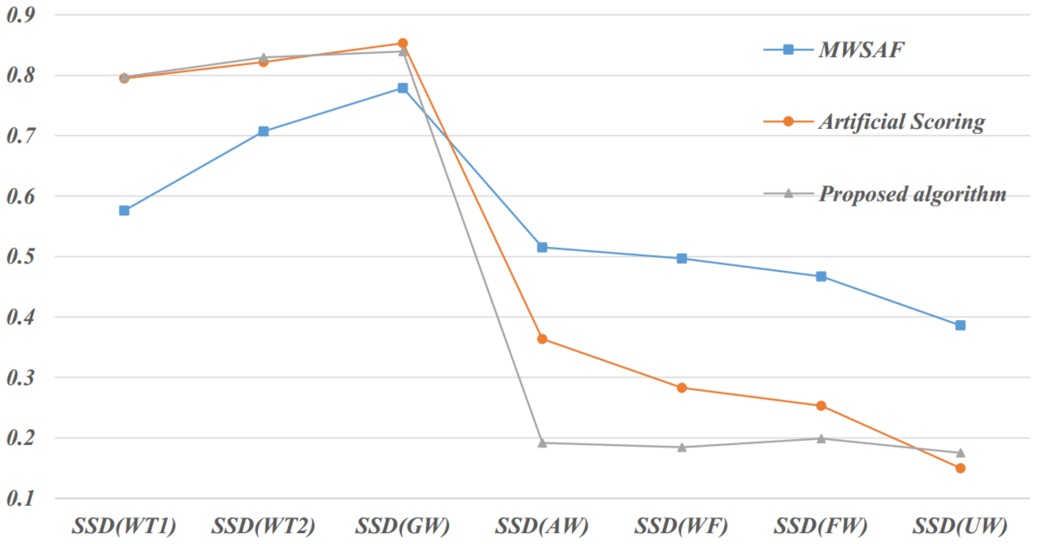


Figure 10. Comparison among artificial scoring, MWSAF, and the proposed algorithm

considers without the internal features that is considered in the proposed approach. This influence is also reflected in the rest four comparisons. MWSAF gives much higher values of the semantic similarity degree when the other two believe there is a lower semantic similarity of the rest four concept pairs. The proposed approach has less deviation than that of MWSAF in the rest four cases. What is more, MWSAF gives more concentrated results that may lead to problem when selecting an OWL concept to annotate a WSDL element. A narrow range of semantic similarity degree values makes a decision on choosing proper target concept from an ontology difficult. Especially, the value of the threshold should be set carefully when the threshold value falls within this narrow range. The proposed approach provides better accuracy, and it provides a clear distinction between the results that makes it easy to select value of threshold.

Table 5 illustrates the acceptance conditions of OWL concept to annotate corresponding WSDL element under different thresholds. For example, concept "*WeatherReport*" in the "*WeatherConcepts.owl*" file can be used to annotate the concept "*WeatherReport*" in both "*WT2.wsdl*" and "*GlobalWeather.wsdl*" documents according to semantic similarity degree measuring of the proposed approach and Artificial Scoring when the threshold is set to be 0.8. If the threshold is 0.9, none concept in "*WeatherConcepts.owl*" can be used to annotate "*WeatherReport*" even if lexical-level similarity is 1.

Table 5. Tree structure mapping rules of OWL

AR \ **A** **T**	**the proposed approach**	**Artificial Scoring**	**MWSAF**
0.1	WT1,WT2,GW,AW,WF,FW,UW	WT1,WT2,GW,AW,WF,FW,UW	WT1,WT2,GW,AW,WF,FW,UW
0.2	WT1,WT2,GW,AW,WF,FW	WT1,WT2,GW,AW,WF,FW	WT1,WT2,GW,AW,WF,FW,UW
0.3	WT1,WT2,GW	WT1,WT2,GW,AW	WT1,WT2,GW,AW,WF,FW,UW
0.4	WT1,WT2,GW	WT1,WT2,GW	WT1,WT2,GW,AW,WF,FW
0.5	WT1,WT2,GW	WT1,WT2,GW	WT2,GW,AW
0.6	WT1,WT2,GW	WT1,WT2,GW	WT2,GW
0.7	WT1,WT2,GW	WT1,WT2,GW	WT2,GW
0.8	WT2,GW	WT2,GW	

* **A:** Approaches; **AR:** Accepted Results under specific Threshold; **T:** Threshold values.

5. Conclusion and Future Work.

In this paper, we have proposed a hybrid semantic similarity measuring approach to implement semantic annotation of legacy Web Services. Firstly, we map concepts in WSDL document and concepts in OWL files to corresponding abstract tree structure. Then, we proposed three internal features, i.e., "depth", "width", and "density", of node in the tree structure based on the previous commonly used approaches, i.e., edge-based, feature-based, and information content-based approaches. At last, we measure lexical-level similarity and structural-level similarity considering the proposed internal features. Analysis and experimental results show that the proposed approach can provide more accuracy value of semantic similarity degree measuring between two concepts from WSDL document and OWL file respectively. In addition, the decision can be made easily to determine which one of the OWL concepts can be used to annotate corresponding WSDL concept. Because the proposed approach obtains semantic similarity degree with high discrimination with a width value range. The proposed approach can also be applied to any other knowledge resources that are written in different description languages.

In the near future, we will utilize "*WordNet*" as the ontology corpus. What is more, we will combine Levenstein Distance and Abbreviation Expansion with synonyms to improve the semantic similarity measuring accuracy from the perspective of lexical-level similarity. Additionally, unused information of the tree structure, such as upper-level nodes and information of edges, will be fully utilized when measuring structural-level similarity.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China (No. 61272353, No. 61370128, No. 61428201, No. 61502028), Program for New Century Excellent Talents in University (NCET-13-0659), Beijing Higher Education Young Elite Teacher Project (YETP0583).

REFERENCES

[1] O. Hatzi, D. Vrakas, M. Nikolaidou, N. Bassiliades, D. Anagnostopoulos and I. Vlahavas, An integrated approach to automated semantic web service composition through planning, *IEEE Trans. Services Computing*, vol.5, no.3, pp.319-332, 2012.

[2] D. Paulraj, S. Swamynathan and M. Madhaiyan, Process model ontology-based matchmaking of semantic web services, *International Journal of Cooperative Information Systems*, vol.20, no.4, pp.357-370, 2011.

[3] M. Klusch, P. Kapahnke, S. Schulte, F. Lecue and A. Bernstein, Semantic web service search: A brief survey, *KI – Künstliche Intelligenz*, vol.30, no.2, pp.139-147, 2016.

[4] T. Berners-Lee, J. Hendler and O. Lassila, The semantic web, *Scientific American*, vol.284, no.5, pp.28-37, 2001.

[5] A. Martinez-Garcia, S. Morris, M. Tscholl, F. Tracy and P. Carmichael, Case-based learning, pedagogical innovation, and semantic web technologies, *IEEE Trans. Learning Technologies*, vol.5, no.2, pp.104-116, 2012.

[6] A. Solé-Ribalta, D. Sánchez, M. Batet and F. Serratosa, Towards the estimation of feature-based semantic similarity using multiple ontologies, *Knowledge-Based Systems*, vol.55, pp.101-113, 2014.

[7] Y. Jiang, X. Zhang, Y. Tang and R. Nie, Feature-based approaches to semantic similarity assessment of concepts using Wikipedia, *Information Processing & Management*, vol.51, no.3, pp.215-234, 2015.

[8] R. Rada, H. Mili, E. Bicknell and M. Blettner, Development and application of a metric on semantic nets, *IEEE Trans. Systems, Man, and Cybernetics*, vol.19, no.1, pp.17-30, 1989.

[9] C. Leacock and M. Chodorow, Combining local context and WordNet similarity for word sense identification, *WordNet: An Electronic Lexical Database*, vol.49, no.2, pp.265-283, 1998.

[10] Y. Li, Z. A. Bandar and D. McLean, An approach for measuring semantic similarity between words using multiple information sources, *IEEE Trans. Knowledge and Data Engineering*, vol.15, no.4, pp.871-882, 2003.

[11] J. B. Gao, B. W. Zhang and X. H. Chen, A WordNet-based semantic similarity measurement combining edge-counting and information content theory, *Engineering Applications of Artificial Intelligence*, vol.39, pp.80-88, 2015.

[12] A. Tversky, Features of similarity, *Psychological Review*, vol.84, no.4, p.327, 1977.

[13] E. G. Petrakis, G. Varelas, A. Hliaoutakis and P. Raftopoulou, X-similarity: Computing semantic similarity between concepts from different ontologies, *Journal of Digital Information Management*, vol.4, no.4, pp.233-237, 2006.

[14] T. Pedersen, S. V. Pakhomov, S. Patwardhan and C. G. Chute, Measures of semantic similarity and relatedness in the biomedical domain, *Journal of Biomedical Informatics*, vol.40, no.3, pp.288-299, 2007.

[15] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, *Proc. of the 14th International Joint Conference on Artificial Intelligence*, vol.1, pp.448-453, 1995.

[16] D. Lin, An information-theoretic definition of similarity, *Internal Conference on Machine Learning*, vol.98, pp.296-304, 1998.

[17] D. Sánchez, M. Batet and D. Isern, Ontology-based information content computation, *Knowledge-Based Systems*, vol.24, no.2, pp.297-303, 2011.

[18] Y. Jiang, W. Bai, X. Zhang and J. Hu, Wikipedia-based information content and semantic similarity computation, *Information Processing & Management*, vol.53, no.1, pp.248-265, 2017.

[19] D. Sánchez and M. Batet, A new model to compute the information content of concepts from taxonomic knowledge, *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol.8, no.2, pp.34-50, 2012.

[20] R. L. Cilibrasi and P. M. Vitanyi, The Google similarity distance, *IEEE Trans. Knowledge and Data Engineering*, vol.19, no.3, pp.370-383, 2007.

[21] N. Seco, T. Veale and J. Hayes, An intrinsic information content metric for semantic similarity in WordNet, *European Conference on Artificial Intelligence*, vol.16, pp.1089-1090, 2004.

[22] Z. Zhou, Y. Wang and J. Gu, A new model of information content for semantic similarity in WordNet, *The 2nd International Conference on Future Generation Communication and Networking Symposia*, vol.3, pp.85-89, 2008.

[23] H. Al-Mubaid and H. A. Nguyen, Measuring semantic similarity between biomedical concepts within multiple ontologies, *IEEE Trans. Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol.39, no.4, pp.389-398, 2009.

[24] M. Batet, S. Harispe, S. Ranwez, D. Sánchez and V. Ranwez, An information theoretic approach to improve semantic similarity assessments across multiple ontologies, *Information Sciences*, vol.283, pp.197-210, 2014.

[25] M. A. Rodriguez and M. J. Egenhofer, Determining semantic similarity among entity classes from different ontologies, *IEEE Trans. Knowledge and Data Engineering*, vol.15, no.2, pp.442-456, 2003.

[26] A. Kilgarriff and C. Fellbaum, *WordNet: An Electronic Lexical Database*, 2000.

[27] A. A. Patil, S. A. Oundhakar, A. P. Sheth and K. Verma, Meteors web service annotation framework, *Proc. of the 13th International Conference on World Wide Web*, pp.553-562, 2004.

[28] D. Canturk and P. Senkul, Semantic annotation of web services with lexicon-based alignment, *IEEE World Congress on Services*, pp.355-362, 2011.

[29] D. Bouchiha and M. Malki, Semantic annotation of web services, *International Conference on Web and Information Technologies*, pp.60-69, 2012.

[30] D. Bouchiha, M. Malki, D. Djaa, A. Alghamdi and K. Alnafjan, Empirical study for semantic annotation of web services, *International Journal of Networked and Distributed Computing*, vol.2, no.1, pp.35-44, 2014.

[31] M. Klein, D. Fensel, F. Van Harmelen and I. Horrocks, The relation between ontologies and XML schemas, *Electronic Trans. Artificial Intelligence*, pp.128-145, 2001.

[32] B. Xu, J. Li and K. Wang, Web service semantic annotation, *Journal of Tsinghua University (Science and Technology)*, vol.46, no.10, pp.1784-1787, 2006.

[33] M. Grcar and D. Mladenic, Visual OntoBridge: Semi-automatic semantic annotation software, *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp.726-729, 2009.

[34] *https://en.wikipedia.org/wiki/Levenshtein_distance.*

[35] D. L. Zhang and T. Lv, Concept frequency-based web service annotation, *Journal of Tongji University (Natural Science)*, vol.6, no.1, pp.103-107, 2008.

[36] A. Tripathy, A. Agrawal and S. K. Rath, Classification of sentiment reviews using n-gram machine learning approach, *Expert Systems with Applications*, vol.57, pp.117-126, 2016.

[37] A. M. Passarotti, J. M. Fitzgerald, J. A. Sweeney and M. N. Pavuluri, Negative emotion interference during a synonym matching task in pediatric bipolar disorder with and without attention deficit hyperactivity disorder – CORRIGENDUM, *Journal of the International Neuropsychological Society*, vol.20, no.3, p.349, 2014.

[38] E. Y. Shinohara, E. Aramaki, T. Imai, Y. Miura, M. Tonoike, T. Ohkuma and K. Ohe, An easily implemented method for abbreviation expansion for the medical domain in Japanese text, *Methods of Information in Medicine*, vol.52, no.1, pp.51-61, 2013.

Appendix A. Original and two modified WSDL documents are presented, and the other source WSDL files used in this paper can be found and downloaded from “*http://lsdis.cs.uga.edu/projects/meteor-s/downloads/*”.

1. Original *“WeatherConcepts.wsdl”* document.

```
<xsd:complexType name=“WeatherReport”>
  <xsd:sequence>
    <xsd:element name=“timestamp” type=“xsd:dateTime”/>
    <xsd:element name=“station” type=“xsd1:Station”/>
    <xsd:element name=“phenomena” type=“xsd1:ArrayOfPhenomenon”/>
    <xsd:element name=“precipitation” type=“xsd1:ArrayOfPrecipitation”/>
    <xsd:element name=“extremes” type=“xsd1:ArrayOfExtreme”/>
    <xsd:element name=“pressure” type=“xsd1:Pressure”/>
    <xsd:element name=“sky” type=“xsd1:Sky”/>
    <xsd:element name=“temperature” type=“xsd1:Temperature”/>
    <xsd:element name=“visibility” type=“xsd1:Visibility”/>
    <xsd:element name=“wind” type=“xsd1:Wind”/>
  </xsd:sequence>
</xsd:complexType>
```

2. "*WT1.wsdl*" adds an element "*situation*" and adjusts only the placement of some elements in the original document "*WeatherConcepts.wsdl*".

```
<xsd:complexType name="WeatherReport">
  <xsd:sequence>
    <xsd:element name="timestamp" type="xsd:dateTime"/>
    <xsd:element name="station" type="xsd1:Station"/>
    <xsd:element name="phenomena" type="xsd1:ArrayOfPhenomenon"/>
    <xsd:element name="situation" type="xsd1:situation"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="situation">
  <xsd:sequence>
    <xsd:element name="precipitation" type="xsd1:ArrayOfPrecipitation"/>
    <xsd:element name="extremes" type="xsd1:ArrayOfExtreme"/>
    <xsd:element name="pressure" type="xsd1:Pressure"/>
    <xsd:element name="sky" type="xsd1:Sky"/>
    <xsd:element name="temperature" type="xsd1:Temperature"/>
    <xsd:element name="visibility" type="xsd1:Visibility"/>
    <xsd:element name="wind" type="xsd1:Wind"/>
  </xsd:sequence>
</xsd:complexType>
```

3. "*WT2.wsdl*" deletes an element "*visibility*" in "*WeatherConcepts.wsdl*".

```
<xsd:complexType name="WeatherReport">
  <xsd:sequence>
    <xsd:element name="timestamp" type="xsd:dateTime"/>
    <xsd:element name="station" type="xsd1:Station"/>
    <xsd:element name="phenomena" type="xsd1:ArrayOfPhenomenon"/>
    <xsd:element name="precipitation" type="xsd1:ArrayOfPrecipitation"/>
    <xsd:element name="extremes" type="xsd1:ArrayOfExtreme"/>
    <xsd:element name="pressure" type="xsd1:Pressure"/>
    <xsd:element name="sky" type="xsd1:Sky"/>
    <xsd:element name="temperature" type="xsd1:Temperature"/>
    <xsd:element name="wind" type="xsd1:Wind"/>
  </xsd:sequence>
</xsd:complexType>
```

Appendix B. "*ArtificialScoring*" of the semantic similarity degree between concept pairs.

Table 6. Semantic similarity of artificial scoring from different volunteers

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**AVG**	**MWSAF** ***(Deviation)***	**Proposed** ***(Deviation)***
SSD(WT1)	0.80	0.86	0.70	0.72	0.90	0.90	0.95	0.90	0.42	0.794	0.576***(0.275)***	0.797***(0.004)***
SSD(WT2)	0.70	0.98	0.60	0.74	0.85	0.90	0.97	0.88	0.77	0.822	0.707***(0.140)***	0.830***(0.010)***
SSD(GW)	0.90	1.00	0.80	0.71	0.80	0.90	0.90	0.80	0.88	0.854	0.779***(0.088)***	0.839***(0.018)***
SSD(AW)	0.60	0.26	0.30	0.06	0.40	0.40	0.50	0.70	0.08	0.367	0.515***(0.403)***	0.192***(0.477)***
SSD(WF)	0.40	0.29	0.30	0.09	0.40	0.20	0.30	0.50	0.03	0.279	0.497***(0.781)***	0.185***(0.337)***
SSD(FW)	0.30	0.42	0.20	0.06	0.30	0.20	0.20	0.50	0.03	0.246	0.467***(0.898)***	0.199***(0.191)***
SSD(UW)	0.10	0.30	0.10	0.00	0.10	0.10	0.05	0.40	0.00	0.128	0.386***(2.016)***	0.175***(0.367)***

*"**Px**": presents the number of a volunteer;*

*"**AVG**": is the average value of semantic similarity of all 9 volunteers;*

*"**(Deviation)**": denotes deviation with respect to the value of artificial scoring.*