An Automated and Fuzzy Approach for Semantically Annotating Services

Supannada Chotipant

Quantum Computation and
Intelligent Systems (QCIS)
School of Software
University of Technology Sydney
(UTS)
Sydney, NSW, Australia
Supannada.Chotipant@student.uts.edu.
au

Farookh Khadeer Hussain

Quantum Computation and
Intelligent Systems (QCIS)
School of Software
University of Technology Sydney
(UTS)
Sydney, NSW, Australia
Farookh.Hussain@uts.edu.au

Omar Khadeer Hussain

School of Business
University of New South Wales
Canberra
Australian Defence Force Academy
ACT, Australia
O.Hussain@adfa.edu.au

Abstract—In the recent past, semantic technologies have played an significant role in service retrieval and service querying. Annotating services semantically enables machines to understand the purpose of services and can further assist in intelligent and precise service retrieval, selection and composition. A key issue in semantically annotating services is the manual nature of service annotation. Manual service annotation requires a large amount of time and updating happens infrequently, hence annotations may get out-of-date due to service description changes. Although some researchers have studied semantic service annotation, they have only focused on web services not business service information. Moreover, their approaches are semi-automated, and still require service providers to select appropriate service annotations. In this paper, we propose a completely automated semantic annotation approach for e-services. The aim of this paper is to semantically annotate a service to relevant service concepts in domain-specific ontologies. Services and service concepts are represented by an extended VSM model, based on fuzzy rules. Then, we link a service to a concept, based on the similarity value of the representing vectors. We found during the experimentation process that the performances of the proposed approach and the VSM-based approach were quite similar and, as a result, developed a system to retrieve services that are annotated to relevant concepts. Experiments using a high service retrieval threshold demonstrated a retrieval approach based on extended VSM annotation performed much better than an approach based on VSM annotation.

Keywords— Semantic Service Annotation; Fuzzy Service Annotation; Fuzzy Service Retrieval

I. INTRODUCTION

Nowadays, many businesses advertise their own service information to customers via web sites. Typically customers search for this information using search engines, like Google or Bing. However these search engines are not designed to search for services and as a result, customers often retrieve a large amount of irrelevant information. When a customer needs to complete a task with more than one service, the work has to be separated into subtasks and search for separately. Because generic systems on the Web do not support a service perspective, a framework that combines and represents business services and knowledge sharing is required [1].

Semantic technology has played an important role in retrieval and service querying, which form a significant part of service-based frameworks. Annotating services semantically enables machines to understand the purpose of services and can further assist in intelligent and precise service retrieval, selection and composition. Currently, the meaning of a service is annotated manually by a service provider and, although very accurate, it is also very time-consuming. Consequently, annotations are often not kept up to date. Because of these limitations, automated service annotation is desirable. Some researchers have studied semantic service annotation for web services, however their work has not included business service information, and it is semi-automated work that requires service providers to choose proper annotations.

In order to semantically annotate business services, the issue of ambiguity needs to be addressed. Typically service information provided over the Internet is described in human language, hence it may be ambiguous due to different senses of words in service description. Consequently, one service information may belong to several service topics with various degrees of memberships. Fuzzy theory has been applied to deal with ambiguous and uncertain work [2] for decades. Nevertheless, based on a thorough review of the existing literature, fuzzy technique has not been investigated for semantic service annotation.

In this paper, we introduce an automated and fuzzy method for semantically annotating services. Business services are annotated by linking to relevant service concepts in a given domain-specific service ontology. Each business service and each service concept are represented by a vector. A service is then linked to a concept if their representing vectors are similar. Inspired by the work of [3, 4] on a FIS-CRM model – a concept-based mechanism for representing documents – we represent services using an Extended VSM-based model. The

proposed model weights terms in a service relating their synonyms. Moreover, we note that the similarity between a service and a service concept is fuzzy and its value is grouped into 3 fuzzy sets such as high, medium and low. Each service may be linked to several service concepts with different degrees of membership function.

To validate our annotation approach, we develop a prototype of service retrieval system in transport service ontology. The system applies a fuzzy-based approach that, when given a query, returns relevant service concepts from a knowledge base. Services that are annotated with returned service concepts are then retrieved. To semantically retrieve relevant concepts, we have applied the ECBR algorithm from [5] to calculate the similarity between a query and each service concept by considering the synonyms of a given query. Similar to annotating services, we note that a query-concept similarity is fuzzy. To overcome the issue of vague similarities, we incorporate fuzzy rules into the service retrieval method.

The outline of this paper is described as follows. Related work about semantic service annotation is shown in Section II. In Section III, to validate our annotation approach, we introduce the proposed fuzzy service annotation system. We present a fuzzy service retrieval approach in Section IV. The experiments and results are demonstrated in Section V, and the conclusion is presented in Section VI.

II. RELATED WORK

The existing research work in semantic service annotation is divided into 3 groups – linguistic-based approach, structured-based approach and hybrid approach.

A. Linguistic-Based Annotation Approach

There exists research approach that annotates services by user-defined key terms. Harald and Mathias propose a folksonomy-based approach for semantically annotating services [6]. Given services, users are required to categorize them with tags. To annotate services, the researchers present a service folksonomy – a graph that consists of vertices of actors, services and tags. The disadvantage of this approach is that annotations may be redundant because users are able to independently tag services. Similar to [6], [7] annotates web services by using crowd sourcing techniques that require a group of people to complete the annotation task.

Much research work focuses on annotating services by matching services with ontologies. Iridescent [8] is a tool that suggests annotations for creating and editing the semantic web service descriptions. It applies common words approach to match elements in web services with ontological concepts. Likewise, Deniz and Pinar apply lexicon-based approach for matching the services and domain-specific ontology [9]. They indicate that the different senses of words should be considered. As a result, synsets in the proposed alignment method are level-sense synsets, not in the form of plain sets.

B. Structure-Based Annotation Approach

There exists research approach that annotates services by considering the similarity between structure of services and ontologies. Bo and Zhiyuan present a semi-automated annotation framework for web services [10]. They also propose a new algorithm to match web service with ontology and then include related ontology in the web service description. To annotate a web service, based on domain-specific ontology, WSDL document – a web service description – is converted to OWL-S format that is registered in the service repository as web service annotation. To discover web services, registered services that are similar to a given service request in OWL-S format are retrieved. The framework combines semantic distance and semantic overlap to compute the service-request similarity.

C. Hybrid Annotation Approach

Some researchers focus on semantically annotating services by considering both keywords and structure of services. Reference [11] proposes a semi-automated annotation framework for web services, called MWSAF. The framework recommends ontologies that are relevant to a given web service document (WSDL file), and users then select appropriate ontologies to be annotations. Both a given web service and ontology are converted to SchemaGraph, and their similarity is combination between ElemMatch – linguistic similarity between service and ontology names – and SchemaMatch – structural similarity between two concepts.

Reference [12] focuses on generating a service network. Given a service, a classification method is called to classify the service into related domain ontologies. The concept similarity between parameter concepts in a given service and ontological concepts in selected domains is then calculated. To calculate the similarity, text similar degree; the similarity between parameter name and concept name, and type similar degree; the similarity between parameter type and ontological property type, are considered. The concept with the highest similarity value is selected to be a service annotation.

Based on a thorough review of the current work, in the field of semantic service annotation, we conclude that 1) much research work has focused on semi-automated annotation approaches for web services, and the current research work lacks developing approaches for automated service annotation of non-web services like online business services, and 2) the existing work has focused on crisp annotation – where each parameter of a web service only belongs to one concept – even though that parameter may partially relate to the other concepts. To address these limitations, we have developed an automated and fuzzy approach for semantically annotating online business services.

III. FUZZY SERVICE ANNOTATION APPROACH

Due to the limitations of existing approaches to service annotation, this paper proposes a new semantic approach that automatically annotates a service to relevant service concepts. For example, a service named "Airlines & Airline Agent Bookings" is annotated to a service concept named "Airline Booking". In this paper, the meaningful information for each

service is presented by the service description entity metadata (SDE metadata) and the conceptualization of services is defined as the service ontology. The proposed annotation approach aims to annotate SDE metadata to relevant service concepts in a domain-specific service ontology. It is noted that one service concept may be linked to many SDEs, while one SDE may link to many service concepts. All SDEs and service ontology are stored in the Service Knowledge Base shown in Fig. 1.
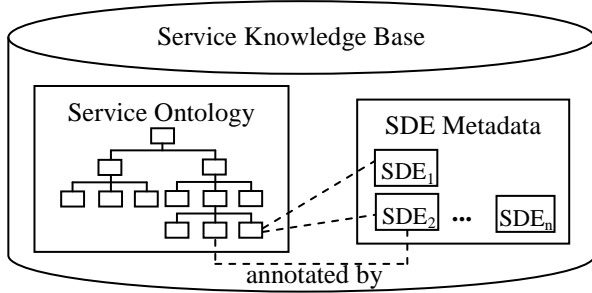

Fig. 1.   Service Knowledge Base

TABLE I.   AN ACTUAL SERVICE CONCEPT 'AIRLINE_AGENT'

Actual Service Concept	
Name	Airline_Agent
Description	Airline Agent, Flight Agent
Linked Metadata	SDE_4267

TABLE II.   SDE METADATA RELATING TO SERVICE CONCEPT 'AIRLINE_AGENT'

SDE_4267 : Airlines & Airline Agent Bookings	
Linked Concepts	Airline_Agent
Provider Name	Virgin Blue Airlines
Provider Address	131 Barry Pde Fortitude Valley QLD, 4006 Australia
Provider Contact Details	Phone No : 13 6789
SDE Description	Low Fares, Great Service.

A.   Service Knowledge Base

The service knowledge base stores information about services and comprises the domain-specific service ontology and SDE metadata. The service knowledge base used in this paper is derived from Hai's work [5]. He conducted a survey of transport service company websites in order to build a transport service ontology. He then crawled company web pages from Yellow Pages in transport category to create SDE metadata.

Service concepts in the service ontology are separated into two groups – abstract concepts and actual concepts. Abstract concepts refer to the abstract domain and the subdomain of

service concepts. Actual concepts relate to real services and can be linked to SDE metadata. Both types of service concepts have two fundamental properties: Service Name and Service Description. Actual service concepts have an additional Linked Metadata property which refers to related SDE metadata.

The SDE metadata for actual business services consists of five properties: relevant service concepts, name of service provider, address of service provider, contact details of service provider, and description of service. An example of an actual service concept and its related SDE is shown in Table I and II respectively.

B.   Fuzzy Service Annotation Approaches

In the service knowledge base proposed in [5], services are essentially connected to actual service concepts according to subcategories that must be defined by service providers. For example, in Table II, the SDE 'Airlines & Airline Agent Bookings' provided by 'Virgin Blue Airlines' is linked to a service concept 'Airline_Agent'. This approach still requires service providers to categorize services into their relevant concepts. This requires a significant amount of time and is therefore updated infrequently. Hence, annotations may become out-dated if the service description changes. In this paper, we introduce new automated method to service annotation called the VSM-based approach and extended VSM-based approach. These approaches semantically link SDE metadata to relevant concepts.

The working process of both approaches is shown in Fig. 2. In the representation module SDE metadata and service concepts in a domain-specific service ontology are represented by vectors. In the matching module, the cosine similarity between vectors of an SDE and a service concept are calculated to find their degree of association. The similarity values range from 0 to 1. If an SDE completely differs from a service concept, the similarity value is 0. In contrast, the value is 1 if an SDE and a service concept are the same. This means that the higher the similarity value, the more closely a SDE relates to a service concept. In this paper, an SDE is linked to a service concept if the similarity value is greater than 0. However, we recognize that the similarity between an SDE and a service concept is fuzzy, and therefore categorize SDE-service concept relationships into three fuzzy sets – high, medium, and low. With a similarity value of s_n, the relationship between an SDE and a concept is high with degree of membership d_i, is medium with a degree of d_j and is low with a degree of d_k.
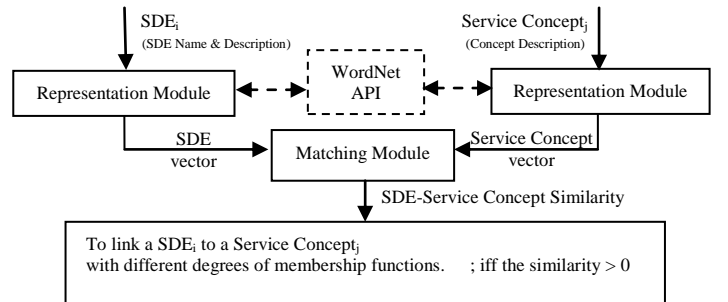

Fig. 2.   Workflow of the service annotation approaches

Although the VSM-based and extended VSM-based approaches appear similar, their representation modules are different. The VSM-based approach creates the representing vectors of SDEs and service concepts using a basic vector space model (VSM). The extended VSM-based approach also includes the semantic of terms of the original VSM. Fig. 2 shows that the dashed box, 'WordNet API', is included in the extended VSM-based approach for the purposes of retrieving synonyms to create a vector. By contrast, the VSM-based approach is not concerned with semantics, and 'WordNet API' is not required. Details of the representation module are described in Section C.

C. Representation Module

The purpose of this module is to generate a vector that represents each SDE or service concept. These vectors are used in the next step to determine the similarities between them.

1) Representation Module for VSM-based Service Annotation Approach

In this section, we present a method to create a vector for use in a vector space model (VSM) - also known as a term vector model – which is typically used to represent text documents. In this paper, we define both SDE metadata and service concepts as text documents, each containing a name and its associated descriptions. We therefore use a VSM model to create vectors for both SDEs and service concepts.

SDEs and service concepts are represented as vectors with N dimensions, in the VSM, where N is the number of terms in the service knowledge base. The value of the vector is the number of times a term occurs in a given document. Vectors for an SDE are created by extracting the SDE name and SDE description as a document. Vectors for a service concept are created by extracting only its concept descriptions, again as a document.

2) Representation Module for Extended VSM-based Service Annotation Approach

Inspired by the work of [3, 4] on a concept-based mechanism for representing documents, we generate a representing vector by considering not only the number of times a term occurs, but also the synonyms for each term in the SDE metadata and service concepts. The resulting extended VSM-based vector is a semantic representation, while VSM-based vector is non-semantic. The main purpose of this module is to increase the weight of a term if its synonyms appear in an SDE or a service concept.

First, the module creates a VSM-based vector for an SDE or a service concept, according to the workflow in the last section. The module, then, connects to the WordNet API to find a set of synonyms for each term. Fig. 2 shows these steps with a dashed box and lines to indicate they are only relevant to the extended VSM-based annotation approach. After retrieving a set of synonyms for each term in the vector, the module readjusts the weight of the term x and its synonyms s_x in an SDE or a service concept as follows:

$$\text{Weight}(x) = C(x) + C(x) * \text{sqrt}(1/T(x))$$

$$\text{Weight}(s_x) = C(s_x) + C(x) * \text{sqrt}(1/T(x))$$

Where: **Weight(x)** is the readjustment weight of term x; **C(x)** is the amount of occurrences of term x; **T(x)** is amount of words having the same meaning as term x; **Weight(s_x)** is the readjustment weight of a synonym for term x; and **C(s_x)** is the amount of occurrences of a synonym of term x. The pseudo-code of the representation module for an extended VSM-based annotation approach is shown as follows.

```
program Representation_Module
   begin
      IF (a document is a SDE metadata) THEN
         terms = noun words from SDE Name and Description
      ELSE IF (a document is a service concept) THEN
         terms = noun words from concept descriptions
      END IF
      Create VSM-based vector
      For each term in terms
         synonyms = get synonyms of term from WordNET API
         IF synonyms is not empty THEN
            For each syn in synonyms
               weight(syn) = C(syn) + C(term)*sqrt(1/T(term))
            END FOR
            weight(term) = C(term) + C(term)*sqrt(1/T(term))
         ELSE weight(term) = C(term)
         END IF
      END FOR
   end.
```

For example, we assume that the term 'storage' occurs three times in the SDE description of SDE_1 and the term 'warehousing' occurs twice in the SDE description of SDE_2. The term 'storage' and the term 'warehousing' are synonyms. The representing VSM-based and extended VSM-based vectors of SDE_1 and SDE_2 are presented in Fig. 3 and Fig. 4 respectively.

	storage	warehousing
SDE_1	3	0

	storage	warehousing
SDE_2	0	2

Fig. 3. VSM-based vectors of SDE_1 and SDE_2

	storage	warehousing
SDE_1	5.121	2.121

	storage	warehousing
SDE_2	1.414	3.414

Fig. 4. Extended VSM-based vectors of SDE_1 and SDE_2

IV. FUZZY SERVICE RETRIEVAL SYSTEM

To validate our annotation approach, we develop a prototype of semantic service retrieval system which enables a user to search services (SDEs) based on service concepts that are relevant to a query. For example, given the query "Flight booking service", the SDE "Airlines & Airline Agent Bookings" provided by Air Niugini is retrieved. The working process of the proposed service retrieval approach is shown in Fig. 5.

The query **Q**, given by a user, is separated into terms and the system finds their synonyms. These terms and their synonyms are input into the extended case based reasoning (ECBR) algorithm to retrieve relevant concepts. The ECBR algorithm computes the value of the similarity between the

query and a service concept. The algorithm adds a value of 1 for each concept description that contains a matching term. A value of 0.5 is added if a matching synonym appears. The similarity value is the matching value divided by the length of the concept description. Where a service concept has more than one concept description, the maximum similarity value is returned. If the similarity value of a service concept is higher than the concept retrieval threshold (CRT), that service concept is selected to find the relevant services in the next step.
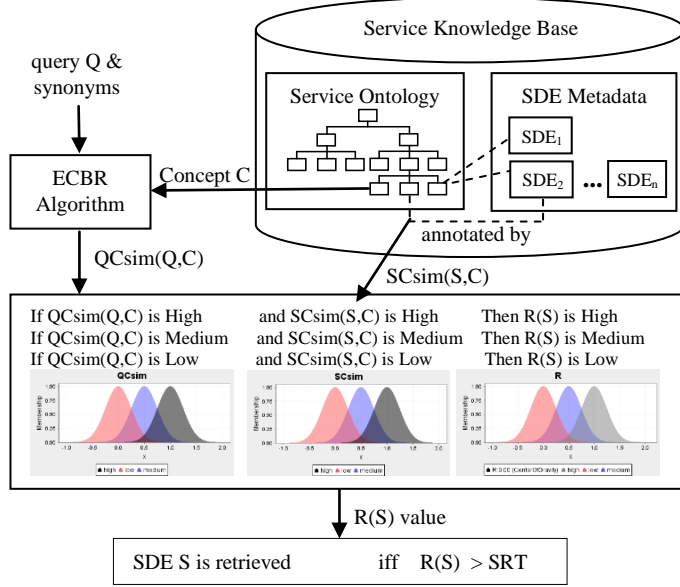

Fig. 5.   Working process of the fuzzy service retrieval approach

In order to retrieve relevant SDEs, we have defined three fuzzy variables – QCsim, SCsim, and R. QCsim is the query-concept similarity value from the ECBR algorithm and SCsim is the service-concept similarity value from the proposed service annotation approach in Section II. R is the retrieval degree for a service. The values of these variables are grouped into 3 levels – high, medium, and low. Additionally, we have defined three simple fuzzy rules that relate to those fuzzy variables as follows.

If QCsim(Q,C) is *High*   **and** SCsim(S,C) is *High*
          Then R(S) is *High*
If QCsim(Q,C) is *Medium* **and** SCsim(S,C) is *Medium*
          Then R(S) is *Medium*
If QCsim(Q,C) is *Low*   **and** SCsim(S,C) is *Low*
          Then R(S) is *Low*

Assuming that QCsim(Q,C) is the similarity between a query **Q** and a service concept **C** and SCsim(S,C) is the similarity between an SDE **S** and a service concept **C**, we indicate that the higher the value of QCsim(Q,C) and SCsim(S,C), the more **S** should be retrieved. Conversely, the lower the value of QCsim(Q,C) and SCsim(S,C), the less **S** should be retrieved. The service retrieval value (**R**) is returned from firing the above fuzzy rules. If the returned retrieval value of a service **S** (R(S)) is greater than the service retrieval threshold (SRT), the service **S** is retrieved from the system.

We implemented the Fuzzy service retrieval system using an open source Java library from [13, 14] called jFuzzyLogic to define the fuzzy membership functions and fuzzy rules, and infer the fuzzy system.

V.   EXPERIMENTS AND RESULTS

In this section, we compare the performance of the service annotation approach based on a VSM model with the approach based on the extended VSM model within the transport domain ontology (TO), as defined in [5]. We also compare the performance of the service retrieval approach based on VSM annotation with the approach based on extended VSM annotation.

In order to evaluate the performance of service annotation and service retrieval, we used four performance measures that are typically applied in information retrieval research field [15] – precision, recall, harmonic mean, and fallout rate. We focused on the retrieved and relevant services of each query when assessing retrieval measures, and on those services of each service concept for annotation measures.

To evaluate the VSM-based and extended VSM-based service annotation approaches, we tested performance by setting the concept retrieval threshold (CRT) for the ECBR algorithm to 0.8. The performance of the VSM-based and extended VSM-based service annotation approaches are shown in Table III. To evaluate the service retrieval approaches, we set the CRT as 0.8, and set the service retrieval threshold (SRT) from 0.1 to 0.9 with an increment of 0.1. Fuzzy retrieval performance was tested with different three types of membership functions – Gaussian, Triangle, and Trapezoidal – as shown in Fig. 6, 7 and 8 respectively. The performance of the VSM-based and extended VSM-based retrieval approaches with these same three functions is presented in Table IV, V and VI. Note that the value of the CRT was set at 0.8 because it gave us the best performance value for precision and recall.
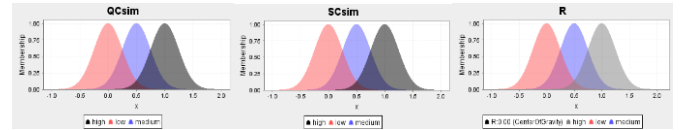

Fig. 6.   Gaussian Membership Functions of QCsim, SCsim, and R variables

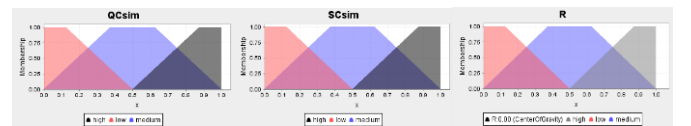Fig. 7.   Triangle Membership Functions of QCsim, SCsim, and R variables

Fig. 8.   Trapezoidal Membership Functions of QCsim, SCsim, and R variables

TABLE III.   TESTING RESULTS OF SERVICE ANNOTATION FOR THE TRANSPORT ONTOLOGY, CRT = 0.8

SRT	Model	Precision	Recall	HM	Fallout
All SRTs	VSM	16.59%	100.00%	10.48%	6.48%
	Extended VSM	14.28%	100.00%	9.24%	7.23%

A. Comparing the performance of VSM-based fuzzy service annotation with Extended VSM-based fuzzy service annotation

Table III demonstrates that the values for precision, harmonic mean and fallout rate for VSM-based service annotations are a little better than those for extended VSM-based service annotation. This is because the extended VSM-based approach takes the semantics of terms in SDEs into consideration and links all SDEs that relate to a concept, even if the similarity value of an SDE and a service concept is low. It is possible that an SDE may be linked to a service concept because only one synonym appears in a service concept description, but its meaning is not related to that concept. By contrast, if we annotate that SDE by VSM-based approach – which focuses on only exactly match –, it is not annotated to that service concept. This leads the precision value of extended VSM based approach to be a little lower. Annotating services with low similarity value causes the precision values to be low – less than 20%. However, this also makes the recall values of both approaches high. It is obviously seen that the recall values are 100%. That is, both approaches can retrieve all relevant SDEs. In conclusion, the experiments showed that the performance of VSM-based and extended VSM-based fuzzy annotation are quite similar.

B. Comparing the performance of VSM-based fuzzy service retrieval with Extended VSM-based fuzzy service retrieval

Based on those three membership functions, it is clear that the values for precision, harmonic mean and fallout rate of VSM-based service retrieval are a little higher than those of extended VSM-based retrieval when the service retrieval threshold (SRT) is less than 0.8. Conversely, the extended VSM-based approach performs much better than VSM-based approach if the SRT is 0.8 or 0.9. For example, in Table IV the precision values of VSM and extended VSM approaches are 57.84% and 85.11% with the setting SRT value as 0.9. That is because the values of extended VSM vector are increased by the readjustment weight from synonyms. Therefore, the values of extended VSM vector tend to be higher than those of VSM vector. As a result, extended VSM-based approach retrieves more SDEs when SRT is high. However, the performance of service retrieval in Table V and VI with the SRT value 0.9 cannot be computed because the approach cannot retrieve any SDE.

Regarding the recall values, they decrease when the SRT values increase. Moreover, it is obviously seen that the recall values of extended VSM model is higher than those of VSM model. In conclusion, the performance of the extended VSM-based retrieval approach is much better than the VSM one on all measures (precision, recall, harmonic mean and fallout rate) when the SRT value is at least 0.8. Moreover, the

Extended VSM-based approach performs better than the VSM one on Recall measure with most SRT values.

TABLE IV.   TESTING RESULTS OF SERVICE RETRIEVAL FOR THE TRANSPORT ONTOLOGY, CRT = 0.8 (GAUSSIAN)

SRT	Model	Precision	Recall	HM	Fallout
0.1	VSM	67.38%	100.00%	37.58%	3.53%
	Extended VSM	63.78%	100.00%	36.39%	3.69%
0.2	VSM	67.38%	100.00%	37.58%	3.53%
	Extended VSM	63.78%	100.00%	36.39%	3.69%
0.3	VSM	67.38%	100.00%	37.58%	3.53%
	Extended VSM	63.78%	100.00%	36.39%	3.69%
0.4	VSM	67.38%	100.00%	37.58%	3.53%
	Extended VSM	63.78%	100.00%	36.39%	3.69%
0.5	VSM	67.56%	100.00%	37.67%	3.41%
	Extended VSM	64.18%	100.00%	36.57%	3.47%
0.6	VSM	87.34%	87.76%	42.36%	1.03%
	Extended VSM	78.28%	80.95%	37.33%	1.02%
0.7	VSM	79.30%	69.74%	35.57%	0.77%
	Extended VSM	78.24%	73.11%	35.81%	0.74%
0.8	VSM	78.17%	53.37%	29.61%	0.39%
	Extended VSM	83.84%	64.57%	33.82%	0.45%
0.9	VSM	57.84%	18.54%	11.66%	0.18%
	Extended VSM	85.11%	32.40%	19.63%	0.22%

TABLE V.   TESTING RESULTS OF SERVICE RETRIEVAL FOR THE TRANSPORT ONTOLOGY, CRT = 0.8 (TRIANGLE)

SRT	Model	Precision	Recall	HM	Fallout
0.1	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.2	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.3	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.4	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.5	VSM	81.69%	59.85%	32.30%	0.60%
	Extended VSM	80.51%	67.36%	33.85%	0.71%
0.6	VSM	82.10%	58.57%	31.88%	0.53%
	Extended VSM	81.36%	66.81%	33.98%	0.49%
0.7	VSM	80.44%	58.21%	31.59%	0.53%
	Extended VSM	81.36%	65.69%	33.31%	0.49%
0.8	VSM	67.64%	14.56%	9.90%	0.14%
	Extended VSM	85.85%	30.74%	19.20%	0.14%
0.9	VSM	-	-	-	-
	Extended VSM	-	-	-	-

TABLE VI.   TESTING RESULTS OF SERVICE RETRIEVAL FOR THE
TRANSPORT ONTOLOGY, CRT = 0.8 (TRAPEZOIDAL)

SRT	Model	Precision	Recall	HM	Fallout
0.1	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.2	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.3	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.4	VSM	81.06%	59.88%	32.04%	0.88%
	Extended VSM	80.30%	67.36%	33.75%	0.86%
0.5	VSM	81.46%	59.85%	32.22%	0.66%
	Extended VSM	80.58%	67.36%	33.89%	0.68%
0.6	VSM	82.10%	58.57%	31.88%	0.53%
	Extended VSM	81.36%	66.81%	33.98%	0.49%
0.7	VSM	80.44%	58.21%	31.59%	0.53%
	Extended VSM	81.36%	65.69%	33.31%	0.49%
0.8	VSM	67.64%	14.56%	9.90%	0.14%
	Extended VSM	85.85%	30.74%	19.20%	0.14%
0.9	VSM	-	-	-	-
	Extended VSM	-	-	-	-

VI. CONCLUSION

In this paper, we introduce a completely automated semantic annotation approach for e-services. The purpose of this paper is to semantically link a service description entity (SDE) to relevant service concepts in domain-specific ontology. SDEs and service concepts are represented by an extended VSM model which readjusts weights in a vector by considering the synonyms of terms in the service and concept descriptions. An SDE is annotated to a service concept if the similarity value of representing vectors is more than zero. We indicate that the relation between a SDE and a service concept is based on defined fuzzy membership functions. Experiments show that the performance of the proposed method and the VSM-based method are quite similar. To validate our annotation approach, we have also developed a fuzzy service retrieval system which retrieves SDEs based on the relevant service concepts. Experiments show that the retrieval approach based on extended VSM annotation performs much better than VSM-based annotation when the service retrieval threshold is high.

In the future, we will improve our research as follows.

1) The extended VSM annotation approach in this paper applies only synonyms to generate a service vector. In the future, we plan to consider other semantic relations, such as hyponyms and homonyms, to improve the performance of service annotation.

2) This paper applies 3 simple fuzzy sets and rules to semantically annotate and retrieve services. In the future, we plan increase a number of fuzzy sets and rules in order to improve the performance of service annotation and retrieval.

3) To validate our annotation approach, this paper provides a prototype of service retrieval system in transport ontology. In the future, we plan to test the retrieval system in other domains.

REFERENCES

[1] F. Nachira, A. Nicolai, P. Dini, M. Le Louarn and L.R. Leon, Digital business ecosystems, European Commission, 2007.

[2] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," Commun. ACM, vol. 37, pp. 77-84, 1994.

[3] J.A. Olivas, P.J. Garcés and F.P. Romero, "An application of the FIS-CRM model to the FISS metasearcher: Using fuzzy synonymy and fuzzy generality for representing concepts in documents.", International Journal of Approximate Reasoning 34, pp. 201-219, 2003.

[4] P.J. Garcés, J.A. Olivas and F.P. Romero, "Conceptual matching in web search using FIS-CRM for representing documents." in EUSFLAT Conf., pp. 63-66. Citeseer, 2003.

[5] H. Dong, F.K. Hussain and E. Chang, "A service search engine for the industrial digital ecosystems.", Industrial Electronics, IEEE Transactions on 58, pp. 2183-2196, 2011.

[6] H. Meyer and M. Weske, "Light-Weight Semantic Service Annotations Through Tagging." in Dan, A., Lamersdorf, W. (eds.) Service-Oriented Computing – ICSOC 2006, vol. 4294, pp. 465-470. Springer Berlin Heidelberg, 2006.

[7] J. Scicluna, C. Blank, N. Steinmetz, and E. Simperl, "Crowd Sourcing Web Service Annotations," in AAAI Spring Symposium: Intelligent Web Services Meet Social Computing, 2012.

[8] T. G. Stavropoulos, D. Vrakas, and I. Vlahavas, "Iridescent: a tool for rapid semantic annotation of web service descriptions," presented at the Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics, Madrid, Spain, 2013.

[9] D. Canturk and P. Senkul, "Semantic Annotation of Web Services with Lexicon-Based Alignment," in Services (SERVICES), 2011 IEEE World Congress on, 2011, pp. 355-362.

[10] B. Jiang and Z. Luo, "A new algorithm for semantic web service matching," Journal of Software, vol. 8, pp. 351-356, 2013.

[11] A.A. Patil, S.A. Oundhakar, A.P. Sheth and K. Verma, "Meteor-s web service annotation framework.", Proceedings of the 13th international conference on World Wide Web, pp. 553-562. ACM, New York, NY, USA, 2004.

[12] H. Wang, Z. Feng, S. Chen, J. Xu and Y. Sui, "Constructing Service Network via Classification and Annotation." in Service Oriented System Engineering (SOSE), Fifth IEEE International Symposium on, pp. 69-73, 2010.

[13] P. Cingolani and J. Alcalá-Fdez, "jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming.", International Journal of Computational Intelligence Systems 6, pp. 61-75, 2013.

[14] P. Cingolani and J. Alcala-Fdez, J, "jFuzzyLogic: a robust and flexible Fuzzy-Logic inference system language implementation." in: FUZZ-IEEE, pp. 1-8. Citeseer, 2012.

[15] R. Baeza-Yates and B. Ribeiro-Neto, Modern information retrieval. ACM press, New York, 1999.