A framework for discovering and classifying ubiquitous services in digital health ecosystems

Hai Dong, Farookh Khadeer Hussain, Elizabeth Chang

Digital Ecosystems and Business Intelligence Institute, Curtin University of Technology, Perth, WA 6845, Australia

ABSTRACT
A digital ecosystem is a widespread type of ubiquitous computing environment comprised of ubiquitous, geographically dispersed, and heterogeneous species, technologies and services. As a sub-domain of the digital ecosystems, digital health ecosystems are crucial for the stability and sustainable development of the digital ecosystems. However, since the service information in the digital health ecosystems exhibits the same features as those in the digital ecosystems, it is difficult for a service consumer to precisely and quickly retrieve a service provider for a given health service request. Consequently, it is a matter of urgency that a technology is developed to discover and classify the health service information obtained from the digital health ecosystems. A survey of state-of-the-art semantic service discovery technologies reveals that no significant research effort has been made in this area. Hence, in this paper, we present a framework for discovering and classifying the vast amount of service information present in the digital health ecosystems. The framework incorporates the technology of semantic focused crawler and social classification. A series of experiments are conducted in order to respectively evaluate the framework and the employed mathematical model.

Received:

Keywords: digital ecosystems, digital health ecosystems, ontology, semantic focused crawlers, semantic service discovery, semantic service classification

1. Introduction

With the emergence of the web and its pervasive intrusion on individuals, organizations, businesses etc., people now realize that they are living in a digital environment analogous to the ecological ecosystem. Consequently, no individual or organization can ignore the huge impact of the web on social well-being, growth and prosperity, or the changes that it has brought about to the world economy, transforming it from a self-contained, isolated, and static pattern to an open, connected, dynamic pattern [7]. Recently, the European Union initiated a research vision in relation to the ubiquitous digital environment, known as the digital ecosystems, with many researchers subsequently focusing on this field [10]. A digital ecosystem is defined as an open, loosely coupled, domain clustered, demand-driven, self-organising agents' environment, where each specie is proactive and responsive for its own benefit or profit [7]. The species are the entities with common interests that participate in the digital ecosystems. These contain biological species such as people, economic species such as organizations and digital species such as software, hardware and applications. The environment refers to the underlying technologies and services that support the digital ecosystems. Additionally, the species are the providers and consumers of the technologies and services. It is obvious that the digital ecosystems constitute a widespread computing environment comprised of heterogeneous, geographically dispersed and ubiquitous species, technologies and services.

From the perspective of services, the species can simultaneously act in the role of service provider and service consumer. Since the digital ecosystems are networked by the heterogeneous, geographically dispersed and ubiquitous biological, economic and digital species, the services published by the species reflect the same features [7]. One direct consequence of the features of the services is that it is difficult for a service consumer to precisely and quickly retrieve a service provider who can provide a requested service. One reason for this problem is that the service information in the web is ambiguous and interspersed with other information such as product information, without a mechanism for service information discovery. Another reason is that the service information in the web is heterogeneous without a mechanism for the classification of the service information [11]. In order to solve the problem, the digital ecosystems propose a service factory by means of which service providers can publish and classify their services when entering the environment [10]. The service factory, however, cannot retrieve and classify the service information that pre-exists in the web, such as the service information in local business directories.

The digital health ecosystems, as a domain within the digital ecosystems, also inherit the similar defects [18]. Health service information in the digital health ecosystems is ambiguous and heterogeneous without sufficient supports for service discovery and classification. The reason for our concern about this domain is that health services contribute to ensuring the health of the main participants within the digital ecosystems – biological and economic species, which plays a crucial role in the stabilization and sustainable development of the digital ecosystems. Therefore, in this research, we are concerned with health service discovery and classification by means of service information disambiguation.

Semantic web, as an ongoing project derived from W3C, facilitates the development of a variety of data exchange formats such as Resource Description Framework (RDF)/Extensible Markup Language(XML) [5, 22], and notations, e.g. RDF Schema (RDFS) [6] and Web Ontology Language (OWL) [24], with the vision of making it possible for the web to understand and satisfy the requests of people and machines using the web content [3]. The notations expressed in the semantic web provide meanings to the web content, which can be used for web information disambiguation. Ontology, as the core of the semantic web, is used to represent specific domain knowledge for knowledge sharing among people and machines [17]. Consequently, we believe that the semantic web notations and ontology can be applied to the digital health domain with the purpose of domain knowledge-based service discovery and classification. However, since one of our goals is to create an automatic service classification methodology, two issues are observed: 1) the service information may be incomplete, thereby affecting the performance of the automatic classification; and 2) the service providers may not agree with the automatic classification results as a result of the differences of individual perceptions towards domain knowledge; in other words, every service provider has its own opinion about how to classify the provided services [12]. To resolve the two issues, we adopt the approach of social classification, which is the process by which a community of users classifies the resources in that community for their own use [23]. In this case, we can invite the service providers to become involved in our service classification process. By means of the social classification, we can obtain leverage between the automatic service classification and the service provider-oriented service classification.

Our research objectives are as follows. We propose to:

- design a methodology for automatic service discovery in the digital health ecosystems;
- design a methodology for domain knowledge-based service classification in the digital health ecosystems; and
- design a platform for service providers to maintain and classify service information.

In order to realize the three objectives above, we propose a novel framework which incorporates a semantic focused crawler and a health service knowledge base for automatic service discovery and classification, as well as a service provider-oriented service classification platform for service provider-oriented service maintenance and classification. The semantic focused crawler is able to discover health service information, to convert the information to semanticized metadata with ontology markup languages, to classify the metadata, and to filter irrelevant metadata with the health service domain knowledge and a classification algorithm. The semantic focused crawler and the classification algorithm are the enhancements incorporated into the previous version of our work [13].

The rest of the paper is organized as follows: in Section 2, we survey the related works in semantic service discovery and semantic focused crawler in order to find out research issues within them; in Section 3, we introduce the system architecture and workflow of the proposed health service discovery and classification framework; in Section 4, we reveal a health service ontology for representing health service domain knowledge, and a health service metadata schema for encapsulating actual heath service information; in Section 5, we depict a mathematical model for realizing the automatic service discovery and classification; in Section 6, we implement a series of experiments for the framework evaluation by means of the traditional information retrieval evaluation metrics; the conclusion is drawn and future work is outlined in the final section.

2. Related works

2.1 Semantic service discovery

The existing semantic service discovery research primarily focuses on web services and other types of e-services.

A number of research efforts in web service discovery have been conducted, which mainly concentrate on semantic annotation of web services for service description disambiguation. The web service environment facilitates several tools for the description and discovery of web services, such as Web Service Description Language (WSDL) [8] and Universal Description, Discovery and Integration (UDDI) [1]. However, in order to discover a web service, the infrastructure should be able to represent the capabilities of a web service, but WSDL and UDDI cannot support such functions. Furthermore, these tools are not able to recognize the similarity between the capabilities being provided and the functionalities being requested [31]. These two challenges highlight the need for semantics for web service discovery. Hence, semantics can be used for describing and reasoning the capabilities of a web service so as to match with the functionalities specified in a service request. This gives rise to the vision of Semantic Web Services (SWS), which integrates the service metadata, domain ontologies, formal tools and web service architecture [25].

In terms of our survey, we classify the emerging researches on semantic web service discovery into four main areas, according to their application environments, including generic, P2P (Peer-to-Peer), Grid, and ubiquitous computing environment. In this paper, we focus only on the literature related to the ubiquitous computing environment, as a result of the similar features shared by the digital ecosystems and the ubiquitous computing environment, which are that the components and services within the two environments are widespread and ubiquitous.

In the ubiquitous computing environment, the resources and access points are highly dynamic [20]. Currently, the researchers in this field primarily focus on Service Discovery Protocol (SDP) development. Vazquez et al. [34] developed a UDP/HTTP-based Multicast Resource Discovery Protocol (mRDP). The mRDP is built on the prerequisite that all resources in the ubiquitous computing environment are annotated with RDF/OWL. The mRDP architecture comprises mRDP clients and mRDP servers. When a semantic powered request is disseminated from an mRDP client to all mRDP servers in the network, each server will model the request with its semantic information models by SPARQL query language (SPARQL) [27], and the Uniform Resource Identifiers (URIs) of the matched resources are returned. Mokhtar et al. [26] developed an Efficient Semantic Service Discovery (EASY) approach, in order to enhance the existing SDPs for the semantic, context-aware and QoS-aware service discovery. The EASY contains an EASY Language (EASY-L) which originates from OWL-S for describing a semantic service, and an EASY Matching (EASY-M) which is a set of conformance relations for matching services in terms of their functional properties and NFPs. EASY can be employed on top of the existing SDPs by adding semantics to their syntactic descriptions. Toninelli et al. [33] provided the framework of a middleware – Adaptable Intelligent Discovery of context-Aware Services (AIDAS), for user-centric semantic service discovery in a mobile environment. AIDAS adds semantics to the properties of interacting entities and the environment by annotating the profiles of services, users and devices. In the middleware, a Discovery Manager (DM) is employed to match service requests and services based on DL-based subsumption reasoning.

Apart from the web service field, some works have been done in other service fields, such as e-services and learning services. Bianchini et al. [4] attempted to use the ontology approach to support e-service publication and discovery. By creating a service provider ontology, a service requester ontology, and three layers of domain service ontology that includes concrete services, abstract services and subject categories, services can be searched by categories, functionalities and a hybrid of context and QoS comparisons. Meanwhile, the semantic relationships between abstract services can be exploited for service discovery. Vega-Gorgojo et al. [35] designed a semantic approach for discovering Computer Supported Collaborative Learning (CSCL) services. They developed an ontology of the CSCL tools, and then introduced an ontology-enabled service discovery infrastructure. In this infrastructure, all CSCL services are stored in the form of semantic metadata. An ontology-compliant reasoner is utilized to enable the retrieval of service instances by a racer query language.

In terms of the analysis of the existing semantic service discovery approaches, we can observe that all of the approaches are built upon the prerequisite that service information needs to be registered and stored in pre-defined registries. In the digital health ecosystems, nevertheless, we cannot intuitively discover service information as it is interspersed with other information. There is no approach provided for discovering specific service information from the web. In addition, there is no domain specific standard for describing services. Last but not least, no research efforts have contributed to the health service domain, so none is available to solve the problem of health service information dispersal and heterogeneity.

2.2 Semantic focused crawlers

In relation to the problems observed above, it can be deduced that an approach for discovering, semantically annotating and classifying the health service information from the web is urgently required. A semantic focused crawler could assist us to solve the problem. Semantic focused crawlers are a subtype of the focused crawlers enhanced by various semantic web technologies with the purpose of crawling web documents under specified topics [14]. The emerging semantic focused crawlers can be primarily classified into two categories as follows:

The first category is ontology-based focused crawlers. These crawlers are able to utilize ontology to classify web documents by computing the similarity values between ontology concepts and descriptions of URLs of web documents [19, 36]. Courseware Watchdog was developed by Tane et al. [32], which has one special feature whereby users can specify their preferences on certain ontology concepts by assigning corresponding weights to the preferred concepts. Then the weights of concepts are aggregated with the similarity values between concepts and web documents in order to obtain user-preferred web documents.

The second category is metadata data abstraction crawlers. These crawlers are able to automatically generate metadata based on web contents by parsing web documents and annotating them with ontology markup languages [15, 16].

In this research, we attempt to integrate the technical specifications of the ontology-focused crawlers and metadata abstraction crawlers in order to discover, annotate and classify health service information in the digital health ecosystems.

3. Service discovery and classification framework for digital health ecosystems

In this section, in order to achieve the three objectives stated in Section 1, we present a Service Discovery and Classification (SDC) framework for the digital health ecosystems, from the perspectives of system architecture and working process respectively.

3.1 System architecture

As presented in Fig. 1, the overall architecture of the SDC framework consists of a semantic focused crawler, a Health Service Knowledge Base and a Service Provider Oriented Health Service Metadata Classifier. In the rest of this section, we will respectively introduce their functions as follows:

The semantic focused crawler is the enhanced version of the model from our previous work [13]. The enhancements include: 1) changing the metadata generation process from single thread to multithread, which improves its crawling capability; 2) removal of the webpage pool for storing web documents and instead directly storing the relevant metadata into the knowledge base, which reduces the requirement for storage space; 3) optimizing the classification algorithm in order to enhance its crawling efficiency. The semantic focused crawler is able to download web documents in order to generate metadata from the downloaded web documents, and to select and classify health service metadata from the generated metadata, which has the following components:

Webpage Fetcher. Its function is to download web documents from a website by a given Uniform Resource Locator (URL) of a website. It is a multithread crawler which runs multiple processes to concurrently download web documents. A series of rules need to be configured before downloading web documents, including the crawling boundary and maximum crawling depth.

Webpage Parser. Its task is to find health service information from the downloaded web documents, and then to parse the health service information documents into information snippets, by referring to actual webpage markup language tags and webpage layouts. In addition, all webpage markup language tags are removed from the web documents.

Metadata Generator. It is employed to generate health service metadata based on the health service information snippets by annotating them with RDF tags. The annotation refers to a unified health service metadata schema, which is discussed in Section 4.2.

Health Service Metadata Classifier. The mission of the classifier is to automatically select and classify health service metadata, by cooperating with the health service domain knowledge stored in the Health Service Knowledge Base. A mathematical model is employed for the classification and filtering, which is introduced in Section 5.

The Health Service Knowledge Base is able to store the upper level health service domain knowledge and the discovered health service metadata, which consists of the following two components:

Health Service Ontology Base. It is used to store an RDFS represented health service ontology (HSO) for the health service metadata discovery and classification. The HSO defines the boundaries of the specific

health service domains, selects and stores service metadata into the knowledge base and classifies them with HSO concepts. The detailed information regarding the HSO is introduced in Section 4.1.

Health Service Metadata Base. It is designed for storing the selected and classified health service metadata. A unified health service metadata schema is included, which is employed to annotate the service metadata.

The aim of the Service Provider Oriented Health Servile Metadata Classifier is to enable health service providers to manually classify and maintain their service metadata, which is comprised of two parts:

Service Provider Messenger. Its mission is to extract the email addresses of service providers from the health service metadata, and send the service providers emails to invite them to login the Service Provider Interface.

Service Provider Interface. The interface provides service providers with the function of maintaining and manually classifying their health service metadata.

3.2 Working process

In this section, we introduce the mechanism of the SDC framework by describing the whole working process as follows:

Step 1. Before the Webpage Fetcher starts to run, its initial visiting URLs, crawling boundary and maximum crawling depth need to be configured. Based on the configuration, the Webpage Fetcher downloads webpages from the web and passes them to the Webpage Parser.

Step 2. On receiving a web document, the Webpage Parser removes all the tags and less important information from the document, and parses it into information snippets in accordance with the predefined heuristics, and passes them to the Metadata Generator. The heuristics need to reference the actual webpage markup language tags and webpage layouts. While we premise that all the downloaded webpages are well-structured, in general, the heuristics are different from website to website, as the information in the webpages of a website normally maintains a consistent style.

Step 3. The Metadata Generator generates metadata based on the information snippets by annotating them with RDF tags.

Step 4. After receiving a metadata from the Metadata Generator, the Health Service Metadata Classifier computes the similarity value between the metadata and each concept in the Health Service Ontology Base. Hence, if there are *n* concepts in a Health Service Ontology Base, the frequency of the similarity computation for a metadata is *n* times. If a similarity value is above a threshold value, the metadata is deemed as relevant to the corresponding concept, and then associated with the concept as well as stored into the Health Service Metadata Base. If there is no association between the metadata and any concepts from the Health Service Ontology Base, the metadata is deemed as irrelevant within the whole health service domain and thus filtered. Hence, the service metadata is classified and filtered by the ontology concepts. The detailed association process is introduced in Section 4.

Step 5. The Service Provider Messenger extracts the email address from a health service metadata and sends an email to the service provider inviting it to participate.

Step 6. On receiving the email, the service provider can use the identification number contained in the email to login the Service Provider Interface to maintain and manually classify the relevant health service metadata.

4. Health service ontology and health service metadata schema

4.1 Health service ontology

As introduced in Section 3.1, an HSO is designed for representing the health service domain knowledge and classifying service metadata, which is stored in the Health Service Ontology Base. The HSO is a four-tier hierarchy of health service concepts linked by the generalization/specification relationship, in which each concept represents a health service sub-domain and inherits the properties from its parent concept. However, because of the knowledge differences among health service subdomains, each concept should have domain-specific properties to outline its specialities. To allow the domain speciality, we define a property of *conceptDescription*, which is an extended property that can be an arbitrary amount. The *conceptDescription* property(s) contains domain-specific information describing a HSO concept, which is used for the forthcoming metadata-concept similarity computation introduced in Section 5. Additionally, to

enable the association between a service concept and a service metadata, we design the property of *linkedMetadata*, which is used to store the URI(s) of metadata associated with a concept. The abbreviated view of a generic HSO concept schema in RDFS can be viewed below:

```
<rdfs:Class rdf:about="& kb;Health_Service"
        rdfs:comment="Health_Service"
        rdfs:label="Health_Service">
        <rdfs:subClassOf rdf:resource="& rdfs;Resource"/>
            <rdf:Property rdf:about="& kb;conceptDescription_1"
                    rdfs:label="conceptDescription">
                    <rdfs:domain rdf:resource="& kb;Health_Service"/>
                    <rdfs:range rdf:resource="& rdfs;Literal"/>
            </rdf:Property>
            …
            <rdf:Property rdf:about="& kb;conceptDescription_#"
                    rdfs:label="conceptDescription">
                    <rdfs:domain rdf:resource="& kb;Health_Service"/>
                    <rdfs:range rdf:resource="& rdfs;Literal"/>
            </rdf:Property>
            <rdf:Property rdf:about="& kb;linkedMetadata"
                    rdfs:label="linkedMetadata">
                    <rdfs:domain rdf:resource="& kb;Health_Service"/>
                    <rdfs:range rdf:resource="& kb;Health_Service_Description_Entity"/>
            </rdf:Property>
</rdfs:Class>
```

4.2 Health service metadata schema

As mentioned in Section 3.1, the semantic focused crawler takes a unified health service metadata schema in order to build the health service metadata. There are two types of metadata schemas involved: the Health Service Description Entity (HSDE) schema and the Health Service Provider (HSP) schema.

The HSDE schema is used to build an HSDE metadata that describes a health service entity, which is defined by the following properties:

healthServiceName. This property provides the name of a service entity.

serviceDescription. Opposite to the *conceptDescription* of the HSO concepts, the *serviceDescription* stores the detailed description of a service entity. Similarly, this property can be an arbitrary amount, which depends on the number of information snippets describing a health service entity. Analogously, this property is used for the similarity computation between a HSDE metadata and a HSO concept.

linkedConcepts. This property is the inverse property of the *linkedMetadata*, which is used to store URIs of relevant concepts in order to realize the association process.

provider. This property is used to reference the relevant HSP metadata by storing their URIs.

The HSDE schema in RDFS is shown below:

```
<rdfs:Class rdf:about="& kb;Health_Service_Description_Entity"
        rdfs:comment="Health_Service_Description_Entity"
        rdfs:label="Health_Service_Description_Entity">
        <rdfs:subClassOf rdf:resource="& rdfs;Resource"/>
            <rdf:Property rdf:about="& kb;healthServiceName"
                    rdfs:label="healthServiceName">
                    <rdfs:domain rdf:resource="& kb;Health_Service_Description_Entity"/>
                    <rdfs:range rdf:resource="& rdfs;Literal"/>
            </rdf:Property>
            <rdf:Property rdf:about="& kb;serviceDescription_1"
                     rdfs:label="serviceDescription">
                    <rdfs:domain rdf:resource="& kb;Health_Service_Description_Entity"/>
                    <rdfs:range rdf:resource="& rdfs;Literal"/>
            </rdf:Property>
            …
            <rdf:Property rdf:about="& kb;serviceDescription_#"
                    rdfs:label="serviceDescription">
```

```
            <rdfs:domain rdf:resource="& kb;Health_Service_Description_Entity"/>
            <rdfs:range rdf:resource="& rdfs;Literal"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;linkedConcepts"
            rdfs:label="linkedConcepts">
            <rdfs:range rdf:resource="& kb;Health_Service"/>
            <rdfs:domain rdf:resource="& kb;Health_Service_Description_Entity"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;provider"
            rdfs:label="provider">
            <rdfs:domain rdf:resource="& kb;Health_Service_Description_Entity"/>
            <rdfs:range rdf:resource="& kb;Health_Service_Provider"/>
        </rdf:Property>
</rdfs:Class>
```

The HSP schema is used to build an HSP metadata that describes a service provider, which consists of the following properties:

providerName. This property is used to store the name of a service provider, e.g. a company name.

providerProfile. This property is used to store the descriptive information about the profile of a service provider.

address. This property is used to store the address information of a service provider.

contactDetails. This property is used to store the contact information of a service provider, including phone number, fax number, URL of website, email etc.

services. *services* is the inverse property of the *provider* property of the HSDE schema, which is used to store the URI(s) of the HSDE metadata relevant to a HSP metadata.

The HSP schema in RDFS is displayed below:

```
<rdfs:Class rdf:about="& kb;Health_Service_Provider"
    rdfs:comment="Health_Service_Provider"
    rdfs:label="Health_Service_Provider">
    <rdfs:subClassOf rdf:resource="& rdfs;Resource"/>
        <rdf:Property rdf:about="& kb;providerName"
            rdfs:label="providerName">
            <rdfs:domain rdf:resource="& kb;Health_Service_Provider"/>
            <rdfs:range rdf:resource="& rdfs;Literal"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;providerProfile"
            rdfs:label="providerProfile">
            <rdfs:domain rdf:resource="& kb;Health_Service_Provider"/>
            <rdfs:range rdf:resource="& rdfs;Literal"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;address"
            rdfs:label="address">
            <rdfs:domain rdf:resource="& kb;Health_Service_Provider"/>
            <rdfs:range rdf:resource="& rdfs;Literal"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;contactDetails"
            rdfs:label="contactDetails">
            <rdfs:domain rdf:resource="& kb;Health_Service_Provider"/>
            <rdfs:range rdf:resource="& rdfs;Literal"/>
        </rdf:Property>
        <rdf:Property rdf:about="& kb;services"
            rdfs:label="services">
            <rdfs:domain rdf:resource="& kb;Health_Service_Provider"/>
               <rdfs:range rdf:resource="& kb;Health_Service_Description_Entity"/>
        </rdf:Property>
</rdfs:Class>
```

From the descriptions of the HSO concept schema, HSDE schema and HSP schema, it can be observed that there are references among the three schemas, in which the HSO concept and HSDE metadata follow a

many-to-many relation, and HSDE metadata and HSP metadata follow a many-to-one relation. These semantic links enable further efficient search applications within the Health Service Knowledge Base from any of the three perspectives. The example of the HSO concept-HSDE metadata-HSP metadata association is presented in Fig. 2.

5. Index-term-based extended case-based reasoning algorithm

As mentioned in Section 3.1, we employ an algorithm for computing the similarity value between an HSDE metadata and an HSO concept. If the similarity value is above a predefined threshold value, the metadata can be determined as being relevant to the concept and thus can be associated with the concept, in order to realize the objective of health domain knowledge-based service discovery and classification. Here we introduce an Index term-based Extended Case-Based Reasoning (IECBR) algorithm, which is the enhanced version of the ECBR algorithm published in our previous papers [12, 13]. By introducing the theory of index terms into the ECBR, it is expected that the IECBR algorithm will be more efficient than the ECBR algorithm.

The principle of the IECBR algorithm is similar to that of the ECBR algorithm, which finds the highest coupling value between the *serviceDescription* property(s) of a HSDE metadata and the *conceptDescription* property(s) of a concept. The theoretical foundation is the belief that a metadata can be defined by its *serviceDescription* property(s), and in parallel, a concept can be defined by its *conceptDescription* property(s). Accordingly, the similarity value between a metadata and a concept can be determined by considering the maximum similarity value between the belonging *serviceDescription* property(s) and *conceptDescription* property(s). The mathematical representation of the IECBR model is shown below:

First of all, a list of index terms k_i *(i=1, 2…n)* is generated from all *conceptDescription* properties *cd* in the HSO. Each cd_j is associated with an array $(w_{1,j}, w_{2,j}...w_{n,j})$ where $w_{i,j} \in \{0, 1\}$ is the weight between k_i and cd_j, 1 indicates that k_i appears in cd_j and 0 indicates no. For each metadata *M*, each of its *serviceDescription* property sd_t is associated with an array $(w_{1,t}, w_{2,t}...w_{n,t})$, where $w_{it} \in \{0, 1\}$ is the weight between k_i and sd_t.

The similarity between an HSDE metadata *M* and an HSO concept *C* and is obtained by Eq. (1) and Eq. (2) as follows:

$$sim(M, C) = \max_{sd_t \in M, cd_j \in C} \left(\sum_{\Omega \in cd_j} \frac{f(sd_t, \Omega)}{\sum cd_j} \right) \tag{1}$$

$$f(sd_t, \Omega) = \begin{cases} 1 \ if \ \exists \Delta \mid (\forall k_i, (g_i(\Omega) = g_i(\Delta)) \wedge (\Delta \in sd_t)) \\ 0 \ otherwise \end{cases} \tag{2}$$

where Ω is the word involved in cd_j, $\sum cd_j$ is the sum of the array $(w_{1,j}, w_{2,j}...w_{n,j})$ associated with cd_j, Δ is the word involved in sd_t, g_i is a function that returns a weight associated with k_i.

6. System evaluation

The system evaluation is mainly divided into two subtasks: 1) evaluating the whole SDC framework; 2) evaluating the employed mathematical model.

6.1 Prototype implementation and evaluation environment setup

According to Hevner et al.[21]'s theory, one of the design science evaluation approaches is to implement a prototype and discover the failures and defects within its functions. Hence, we implement the whole SDC framework in three steps corresponding to the three parts involved as follows:

First of all, we build the prototype of the semantic focused crawler with Java. To evaluate its functions, we run the crawler to crawl business webpages under the category of health care services in the Kompass website (http://www.kompass.com). The crawler downloads 1800 webpages and generates 1711 metadata in total. This experiment preliminarily proves the validity of the crawler, and a further evaluation of the crawler combined with an evaluation of the IECBR algorithm is described in Section 6.3.

Second, we use Java and Jena API to build the Health Service Knowledge Base. RDFS is used for building the HSO schema, HSDE schema and HSP schema. RDF is employed as a markup language to annotate metadata according to each schema. There are 218 concepts in the HSO.

Thirdly, we build an online platform for realizing the Service Provider Oriented Health Service Metadata Classifier with Java, JavaScript, AJAX, MySQL and HTML. A screenshot of the Service Provider Interface is shown in Fig. 3. When a service provider logs into the Service Provider Interface, he/she is allowed to modify the properties of the HSDE metadata and the properties of the HSP metadata that belong to him/her. One key function is that service providers can modify the *linkedConcepts* property of the metadata. To realize the function, we design a lightweight search engine to allow service providers to search preferred concepts from the HSO. By ticking the preferred concepts from the retrieved concept collection, the metadata can be associated with the chosen concepts. Thus, we preliminarily realize the objective of the service provider-oriented service metadata classification.

According to the above descriptions and screenshot, it can be concluded that the whole framework basically realizes its proposed functions, which preliminarily proves its feasibility. In the next two sections, we evaluate the performance of the IECBR algorithm in the semantic focused crawler for service discovery and classification.

6.2 Performance indicators

In order to thoroughly evaluate the performance of our IECBR algorithm, we employ eight indicators from the field of information retrieval, which are: harvest rate, precision, mean average precision, recall, F-measure, F-measure$_\beta$, fallout rate and crawling time. Here we provide their definitions for the forthcoming experiments.

Harvest rate in the information retrieval is used to measure the crawling ability of a crawler. In this experiment, harvest rate is the proportion of associated metadata in the whole collection of generated metadata, which can be mathematically represented as:

$$\text{Harvest rate} = \frac{\text{Number of associated metadata}}{\text{Number of generated metadata}} \tag{3}$$

Precision in the information retrieval is used to measure the preciseness of a retrieval system [2]. In this experiment, precision for a single concept is the proportion of the relevant metadata associated by this concept in all the metadata associated by this concept, which can be mathematically represented as:

$$\text{Precision(S)} = \frac{\text{Number of associated and relevant metadata}}{\text{Number of associated metadata}} \tag{4}$$

With regard to the whole collection of concepts in an ontology, the precision is the sum of the precision value for each concept normalized by the number of concepts in the collection, which can be represented as:

$$\text{Precision(W)} = \frac{\sum_{i=1}^{n} \text{Precision}(S_i)}{n} \tag{5}$$

Before we introduce the definition of mean average precision, the concept of average precision should be defined. Average precision for a single concept is the average of precision values after truncating a ranked metadata list associated by this concept after each of the relevant metadata for this concept [2]. This indicator emphasizes the return of more relevant metadata earlier, which can be represented as:

$$\text{Average precision(S)} = \frac{\text{Sum(Precision @ Each relevant metadata in the list)}}{\text{Number of associated and relevant metadata in the list}} \tag{6}$$

Mean average precision refers to the average of the average precision values for the collection of concepts in an ontology, which can be represented as:

$$\text{Mean average precision} = \frac{\sum_{i=1}^{n} \text{Average precision}(S_i)}{n} \tag{7}$$

Recall in the information retrieval refers to the measure of effectiveness of a query system [2]. In this experiment, recall for a single concept is the proportion of the relevant metadata associated by this concept in all the relevant metadata of this concept in the collection of generated metadata, which can be represented as:

$$\text{Recall}(S) = \frac{\text{Number of associated and relevant metadata}}{\text{Number of relevant metadata}} \tag{8}$$

With regard to the whole collection of concepts in an ontology, the recall value is the sum of the recall value for each concept normalized by the number of concepts in the collection, which can be represented as:

$$\text{Recall}(W) = \frac{\sum_{i=1}^{n} \text{Recall}(S_i)}{n} \tag{9}$$

It is important to note that the number of relevant metadata can be determined only by a peer-reviewed method, as the estimation of relevance between metadata and concept requires a detailed knowledge of all concepts and metadata in the knowledge base, which can only be manually implemented in the current situation.

F-measure in the information retrieval is used as an aggregated performance scale for a search system [2]. In this experiment, F-measure value is the mean of precision value and recall value, which can be represented as:

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \tag{10}$$

When the F-measure reaches the highest point, it means the integrated value between precision and recall reaches the highest at the same time.

F-measure_β is another measure that combines precision and recall, and the difference is that users can specify the preference on recall or precision by configuring different weights [28]. In this experiment, we employ F-measure (β=2) that weights recall twice as much as precision, which is close to the fact that most search engines are more concerned with recall other than precision, as a result of most users' purposes in obtaining information [30]. The F-measure (β=2) can be represented below as:

$$\text{F-measure } (\beta=2) = \frac{(1+\beta^2) \cdot \text{Precision} \times \text{Recall}}{\beta^2 \cdot \text{Precision+Recall}} = \frac{5 \times \text{Precision} \times \text{Recall}}{4 \times \text{Precision+Recall}} \tag{11}$$

All of the above indicators have the same limitation – they do not consider the amount of irrelevant metadata in an associated metadata collection. Furthermore, if there is no relevant metadata in the associated collection, Recall cannot be defined. To resolve this issue, we need another performance indicator – fallout rate [2]. In this experiment, fallout rate for a single concept is the proportion of irrelevant SDE metadata associated by this concept within the whole collection of irrelevant metadata for this concept in the generated metadata, which can be represented as:

$$\text{Fallout rate}(S) = \frac{\text{Number of associated and irrelevant metadata}}{\text{Number of irrelevant metadata}} \tag{12}$$

With regard to the whole collection of concepts, the fallout rate is the sum of the fallout rate for each concept normalized by the number of concepts in an ontology, which can be represented as:

$$\text{Fallout rate(W)}=\frac{\sum_{i=1}^{n}\text{Fallout rate}(S_i)}{n} \tag{13}$$

In contrast to other performance indicators, the lower the fallout value, the better is the crawler's performance.

Crawling time is an important metrics for evaluating the efficiency of a crawler, which is defined as the interval between the time of reading a web document and the time of classifying all the metadata generated from the web document.

6.3 Latent semantic indexing algorithm

In order to horizontally evaluate the performance of the IECBR algorithm, we adopt a Latent Semantic Indexing (LSI) algorithm, which is a classical algorithm used for information retrieval and document classification [9], in our semantic focused Crawler, for service discovery and classification. The implementation details are as follows:

In the Health Service Knowledge Base, first of all, each HSO concept *C* is regarded as a body of plain texts comprised of *conceptDescription* property(s). Following that, an index term list is obtained from all the concepts in the HSO. Based on the index term list, each HSO concept *C* is formed as an array in which each element is obtained by the term frequency-inverse document frequency (tf-idf) [29], and all the concepts in the HSO are formed as a term-concept matrix *A*. The term-concept matrix is then decomposed by the Singular Value Decomposition (SVD) approach, which can be mathematically represented by Eq. (14):

$$A = U\Sigma V^T \tag{14}$$

where *U* is the matrix derived from the term-to-term matrix given by AA^T, V^T is the matrix derived from the transpose of the concept-to-concept matrix given by A^TA, and Σ is a $r\times r$ diagonal matrix of singular values where $r = min(t, N)$ is the rank of *A*.

Considering now that only *k* largest singular values of Σ are kept along with their corresponding columns in *U* and V^T, the resultant A_k matrix is the matrix of rank *k* which is closest to the original matrix *A* in the least square sense. This matrix is given by Eq. (15):

$$A_k = U_k\Sigma_k V_k^T \tag{15}$$

where $k\ (k < r)$ is the dimensionality of a reduced concept space.

Analogous to the HSO concept, a HSDE metadata *M* can be regarded as a body of plain texts comprised its *serviceDescription* property(s). The HSDE metadata *M* can thus be formed as an index term-based array in which each element is the tf-idf weight between the metadata and a term from the index term list. The array can then be translated into the concept space by Eq. (16), and then compared with A_k by the cosine algorithm, in order to calculate the similarity value between the HSDE metadata *M* and the HSO concept *C*, which can be represented by Eq. (17):

$$M' = \Sigma_k^{-1}U_k^T M \tag{16}$$

$$sim(C,M) = \frac{A_k \cap M'}{|A_k|\times|M'|} \tag{17}$$

Similar to the IECBR model, an optimal threshold value needs to be determined for the LSI model in order to decide whether or not the pairwise metadata and concept are relevant.

6.4 System evaluation

As mentioned in Section 6.1, we choose the Kompass website as the testing data source; the parameters of the testing data and the prototype can be seen in Table 1.

Following that, we respectively run the IECBR and LSI, in order to examine their performance on service discovery and classification, based on the eight indicators.

As described in Section 3.2, after the similarity value between a metadata and a concept is obtained by the IECBR algorithm, a threshold value needs to be decided in order to determine whether or not the metadata and concept should be associated. Consequently, there is another task involved in the evaluation, which is to find the optimal threshold value on which the crawler can gain the best performance. In order to acquire the optimal threshold value, we set the initial threshold value to 0.5, and set the eventual threshold value to 1, with an increment of 0.05 each time. After that, we obtain the performance data of both the IECBR and LSI on each time of the threshold value variation. The evaluation results are shown in Fig. 4 to Fig. 11.

Fig. 4 depicts the performance of the IECBR and LSI on harvest rate. With the increase of the threshold value, the harvest rate for both the algorithms experiences a gradual fall, since the higher threshold value may present a barrier to the association between metadata and concepts. As a whole, the IECBR is more stable, ranging from 98.83% to 84.63%, compared with the LSI that ranges from 91.70% to 19.11%. This experiment proves that the crawling ability of the IECBR can resist the influence of the threshold value variation to some degree.

Fig. 5 shows the performance of the IECBR and LSI on precision. Opposite to the trends in harvest rate, their precision values increase when the threshold value increases, since the higher threshold may filter more non-relevant metadata. The precision value of the IECBR is higher than that of the LSI in most of the intervals except for the two ends. There is a 21.04% gap between the IECBR and LSI on average, which reveals that the IECBR is able to precisely match a metadata with a concept.

Fig. 6 displays the performance of the IECBR and LSI on mean average precision. The curves of the two algorithms on this indicator are almost parallel to their curves on precision, and the IECBR is 20.04% higher than the LSI on average.

Fig. 7 reveals the performance of the IECBR and LSI on recall. There is a 40% gap between the IECBR and LSI in most of the intervals, which indicates that the IECBR has an overwhelming advantage over the LSI on the effectiveness. Moreover, even at the extreme threshold value (1.0), the IECBR maintains its recall value in a relatively stable position (89.68%), which shows that the IECBR has a relatively stronger ability to retrieve more relevant metadata.

As an aggregated metrics, the F-measure values of the IECBR and LSI are shown in Fig .8. The result shows that the IECBR is more outstanding than the LSI, as a result of the advantages in both precision and recall. The average F-measure value for the IECBR is 82.04%, compared with 48.66% for the LSI.

F-measure (β=2) places twice the weight on recall than on precision, which is depicted in Fig. 9. Due to the significant advantage on recall, the gap between IECBR and LSI on F-measure (β=2) is larger than for the F-measure, which is 37.5% on average.

Fallout rate shows the error rates of the two algorithms, which are displayed in Fig. 10. Similarly, in most of the intervals, the fallout rate for IECBR is less than for the LSI except for the threshold values of 0.5 and 1.

Crawling time is a key parameter for examining the efficiency of the algorithms. Fig. 11 demonstrates the performance of the IECBR, ECBR and LSI on crawling time. It is observed that, with the rise in number of crawled webpages, the IECBR uses less time than do the other two algorithms. The statistics show that IECBR is nearly 44% more efficient than the ECBR, and nearly 160% more efficient than the LSI.

As a conclusion to the evaluation, all the experiments show that the IECBR performs better than the LSI, along with the threshold value variations. This indicates that the IECBR is more adaptable to the environment of a semantic focused crawler than is the LSI. Consequently, we can conclude that the feasibility of the IECBR is preliminarily proven by this series of experiments. Furthermore, in the crawling time test, the IECBR shows higher efficiency than the ECBR. Thus, we achieve our goal of modifying the ECBR algorithm, the aim of which is to maintain the same performance with a lower computing cost.

For the task of the optimal threshold value selection, since F-measure and F-measure (β=2) are two aggregated metrics, we place more weight on them when deciding the threshold value. Thus, by referring to Figs. 8 and 9, the optimal threshold values for the highest F-measure and F-measure (β=2) are both 0.7. The performance of the IECBR on this threshold value is shown in Table 2.

7. Conclusion and future works

In this paper, we deliver a SDC framework for service discovery and classification in the health digital ecosystems. The framework consists of three parts as follows:

- A semantic focused crawler for automatic health service information discovery, health service metadata generation and classification. The discovery and classification is based on associating the health service metadata with the similar HSO concepts.
- A Health Service Knowledge Base for providing the HSO in order to assist the crawler to discover and classify the health service metadata, and for storing the discovered HSDE and HSP metadata.
- A Service Provider Oriented Health Service Metadata Classifier which enables service providers to maintain and manually classify the health service metadata in order to achieve the agreement between the automatic metadata classification and the service provider-based metadata classification.

In addition, we design a more efficient IECBR algorithm, an enhanced version of the ECBR algorithm,, in order to compute the similarity values between the HSDE metadata and the HSO concepts as the basis for determining whether the pair-wise metadata and concept should be associated. In order to evaluate the framework, we implement a prototype, and adopt the method of functional testing to evaluate whether the prototype realizes the proposed functions. On the other side, we evaluate the IECBR algorithm by comparing its performance with an LSI algorithm on eight performance indicators from the information retrieval field. Both of the evaluations reveal positive results, which preliminarily prove the feasibility of our research.

For future work, since a further evaluation of the SDC framework needs the involvement of health service providers, we propose a series of call-for-participation activities and subsequent social surveys in order to evaluate the actual impact of the framework on the health service domain. Additionally, we are developing a series of search applications for the Health Service Knowledge Base, in order to allow a service consumer with little relevant domain knowledge to precisely retrieve a reliable service, based on the technology of semantics and QoS.

Reference

[1] Introduction to UDDI: Important features and functional concepts, available at http://www.uddi.org.
[2] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison-Wesley, New York, 1999.
[3] T. Berners-Lee, The semantic web, Scientific American Magazine (2001).
[4] D. Bianchini, V. De Antonellis, B. Pernici, P. Plebani, Ontology-based methodology for e-service discovery, Inf.. Syst. 31 (2006) 361-380.
[5] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau, Extensible Markup Language (XML) 1.0 (Fourth Edition)-origin and goals, available at http://www.w3.org/TR/2006/REC-xml-20060816/.
[6] D. Brickley, R. V. Guha, RDF Vocabulary Description Language 1.0: RDF Schema, available at http://www.w3.org/TR/rdf-schema/.
[7] E. Chang, M. West, Digital ecosystems and comparison to existing collaboration environment., WSEAS T. Environ. and Dev. 2 (2006) 1396-1404.
[8] E. Christensen, F. Curbera, G. Meredith, S. Weerawarana, Web Services Description Language (WSDL) 1.1, available at http://www.w3.org/TR/wsdl.
[9] S. C. Deerwester, S. T. Dumais, G. W. Furnas, R. A. Harshman, T. K. Landauer, K. E. Lochbaum, L. A. Streeter, Computer information retrieval using latent semantic structure. vol. 4839853, U. S. Patent, Ed. USA: Bell Communications Research, Inc., 1988.
[10] P. Dini, N. Rathbone, M. Vidal, P. Hernandez, P. Ferronato, G. Briscoe, S. Hendryx, The digital ecosystems research vision: 2010 and beyond, Creative Commons, 2005.
[11] H. Dong, F. K. Hussain, E. Chang, A QoS-based service retrieval methodology for digital ecosystems, Int. J. Web Grid Serv. 5 (2009) 261-283.
[12] H. Dong, F. K. Hussain, E. Chang, A hybrid service metadata clustering methodology in the Digital Ecosystem environment, in: The IEEE 23rd International Conference on Advanced Information Networking and Applications, 2009, pp. 238-243.
[13] H. Dong, F. K. Hussain, E. Chang, Focused crawling for automatic service discovery, annotation and classification in industiral digital ecosystems, IEEE T. Ind. Electron. Submitted.
[14] H. Dong, F. K. Hussain, E. Chang, State of the art in semantic focused crawlers in: Computational Science and Its Applications – ICCSA 2009, 2009, pp. 910-924.
[15] E. Francesconi, G. Peruginelli, Searching and retrieving legal literature through automated semantic indexing, in: ICAIL '07, 2007, pp. 131-138.
[16] C. L. Giles, Y. Petinot, P. B. Teregowda, H. Han, S. Lawrence, A. Rangaswamy, N. Pal, eBizSearch: A niche search engine for e-business, in: SIGIR'03, 2003, pp. 213-214.
[17] T. Gruber, A translation approach to portable ontology specifications, Knowl. Acquis. 5 (1995) 199-220.
[18] M. Hadzic, A. Sidhu, Digital Health Ecosystems, in: 2nd IEEE International Conference on Digital Ecosystems and Technologies, 2008, pp. cv-cvii.
[19] M. Halkidi, B. Nguyen, I. Varlamis, M. Vazirgiannis, THESUS: organizing web document collections based on link semantics, VLDB J. 12 (2003) 320–332.
[20] U. Hansmann, L. Merk, M. S. Nicklous, T. Stober, Pervasive computing: The mobile world, 2 ed., Springer-Verlag, New York, 2003.

[21] A. R. Hevner, S. T. March, J. Park, S. Ram, Design science in information systems research, MIS Quart. 28 (2004) 75-105.
[22] G. Klyne, J. J. Carroll, Resource Description Framework (RDF): concepts and abstract syntax, available at http://www.w3.org/TR/rdf-concepts/.
[23] A. Mathes, Folksonomies: Cooperative classification and communication through shared metadata., University of Illinois Urbana-Champaign, Champaign, 2004.
[24] D. L. McGuinness, F. v. Harmelen, OWL Web Ontology Language: Overview, available at http://www.w3.org/TR/owl-features/.
[25] S. McIlraith, D. Martin, Bringing semantics to web services, IEEE Intell. Syst. 18 (2003) 90–93.
[26] S. B. Mokhtar, D. Preuveneers, N. Georgantas, V. Issarny, Y. Berbers, EASY: Efficient semAntic Service discoverY in pervasive computing environments with QoS and context support, J. Syst. Softw. 81 (2008) 785-808.
[27] E. Prud'hommeaux, A. Seaborne, SPARQL Query Language for RDF, available at http://www.w3.org/TR/rdf-sparql-query/.
[28] C. J. v. Rijsbergan, Information Retrieval, Butterworths, London, 1979.
[29] K. Spärck Jones, A statistical interpretation of term specificity and its application in retrieval, J. Doc. 28 (1972) 11-21.
[30] L. T. Su, The relevance of recall and precision in user evaluation, J. Am. Soc. Inf. Sci. 45 (1999) 207-217.
[31] K. Sycara, M. Paolucci, A. Ankolekar, N. Srinivasan, Automated discovery, interaction and composition of Semantic Web services, J. Web Semant. 1 (2003) 27-46.
[32] J. Tane, C. Schmitz, G. Stumme, Semantic resource management for the web: an elearning application, in: WWW2004, 2004, pp. 1-10.
[33] A. Toninelli, A. Corradi, R. Montanari, Semantic-based discovery to support mobile context-aware service access, Comput. Commun. 31 (2008) 935-949.
[34] J. I. Vazquez, D. Lo´pez-de-Ipin˜a, mRDP: An HTTP-based lightweight semantic discovery protocol, Comput. Netw. 51 (2007) 4529-4542.
[35] G. Vega-Gorgojo, M. L. Bote-Lorenzo, E. G´omez-S´anchez, Y. A. Dimitriadis, J. I. Asensio-P´erez, A semantic approach to discovering learning services in grid-based collaborative systems, Future Gener. Comp. Syst. 22 (2006) 709-719.
[36] M. Yuvarani, N. C. S. N. Iyengar, A. Kannan, LSCrawler: a framework for an enhanced focused web crawler based on link semantics, in: the 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI'06), 2006, pp. 794-800.

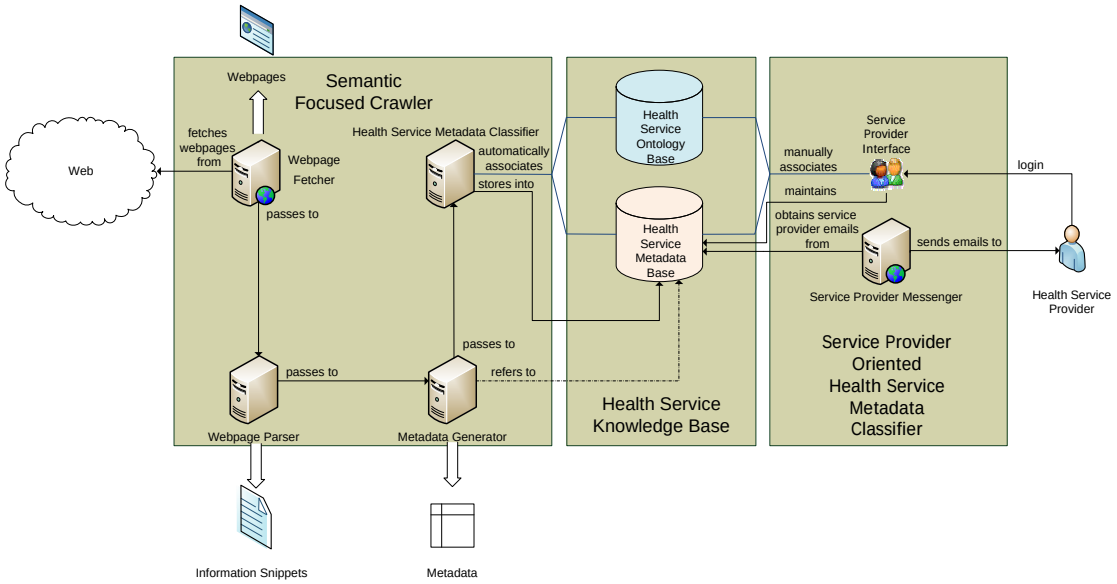

Fig. 1. System architecture of SDC framework for digital health ecosystems

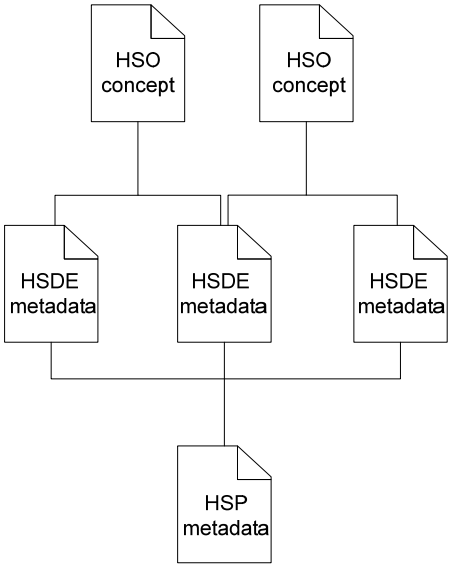

Fig. 2. Example of HSO concept-HSDE metadata-HSP metadata association

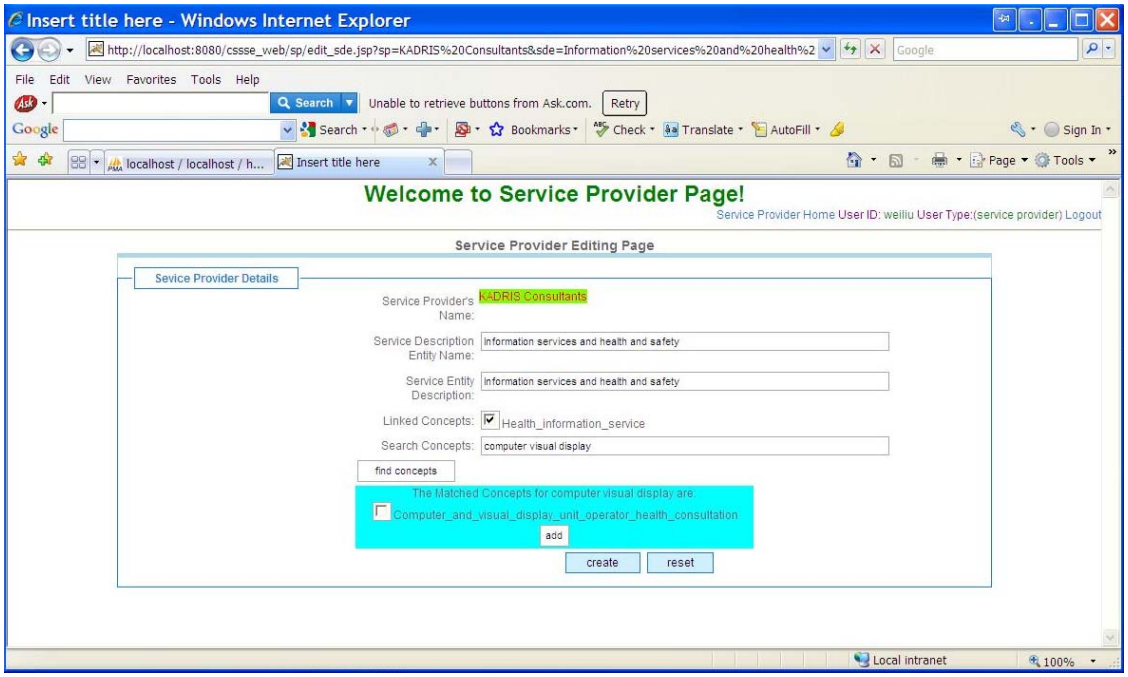

Fig. 3. Screenshot of Service Provider Interface

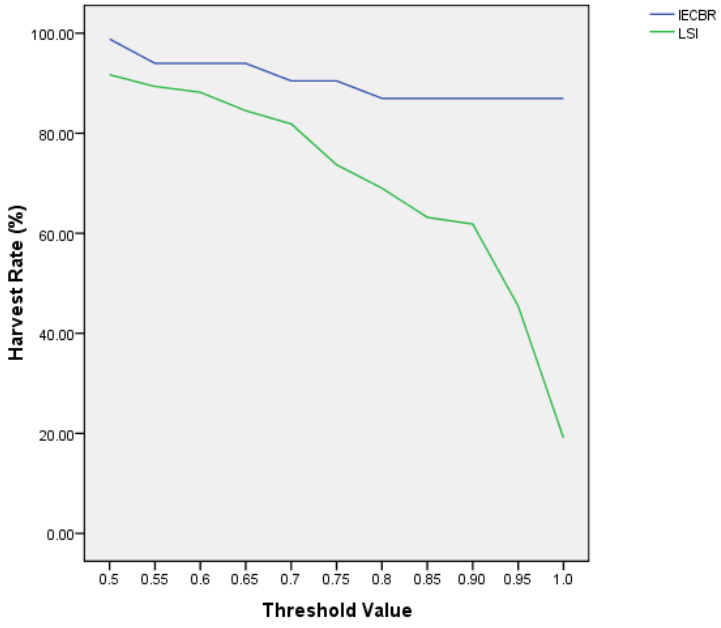

Fig. 4. Comparison of harvest rate between IECBR and LSI

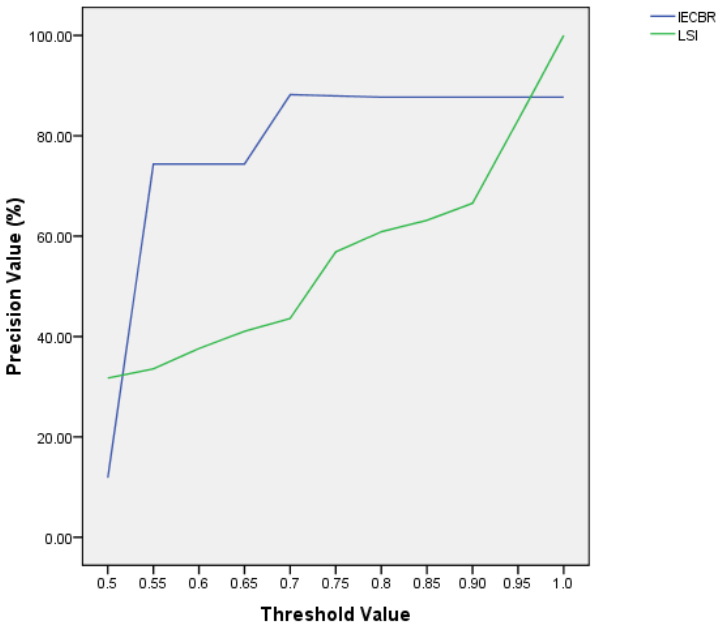

Fig. 5. Comparison of precision between IECBR and LSI

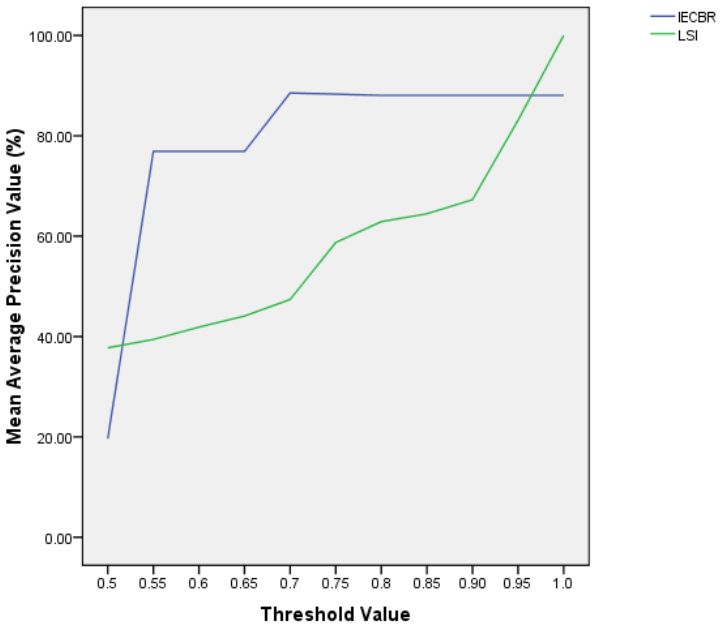

Fig. 6. Comparison of mean average precision between IECBR and LSI

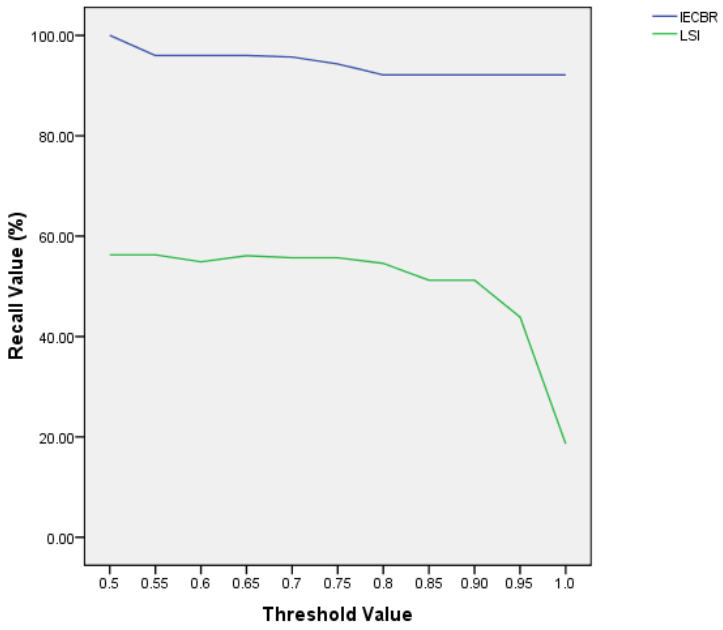

Fig. 7. Comparison of recall between IECBR and LSI

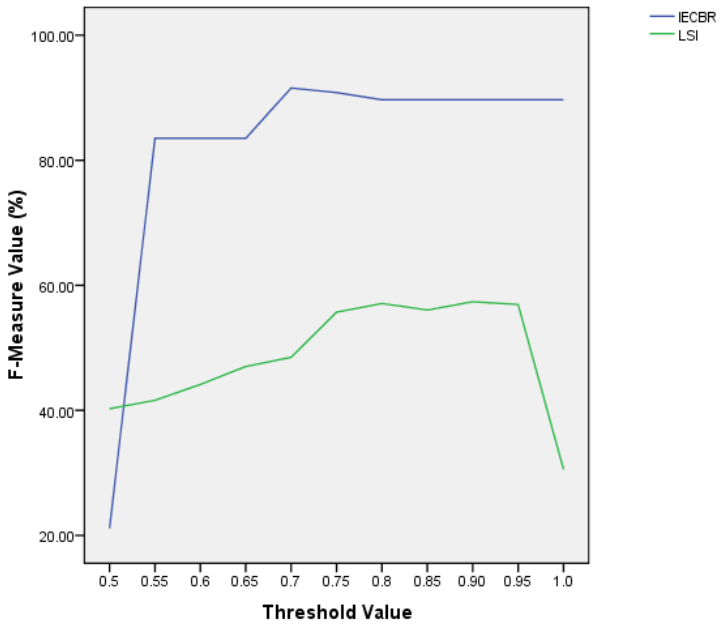

Fig. 8. Comparison of F-measure between IECBR and LSI

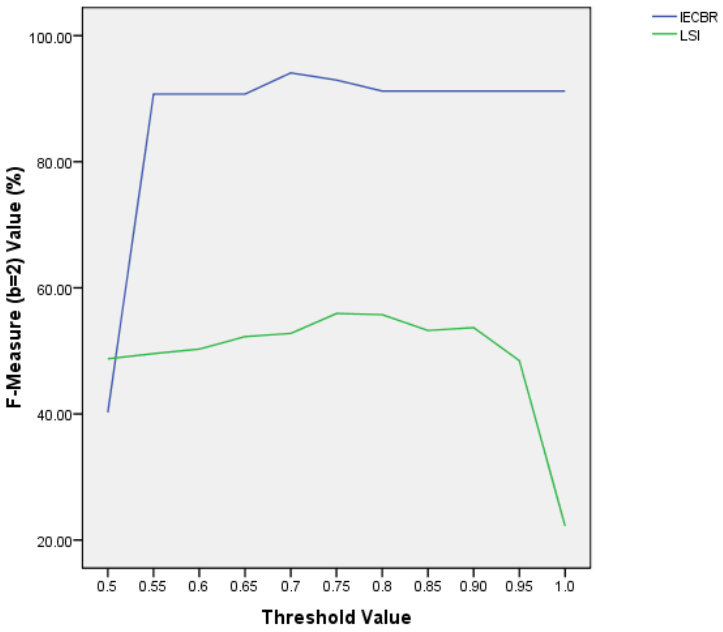

Fig. 9. Comparison of F-measure (β=2) between IECBR and LSI

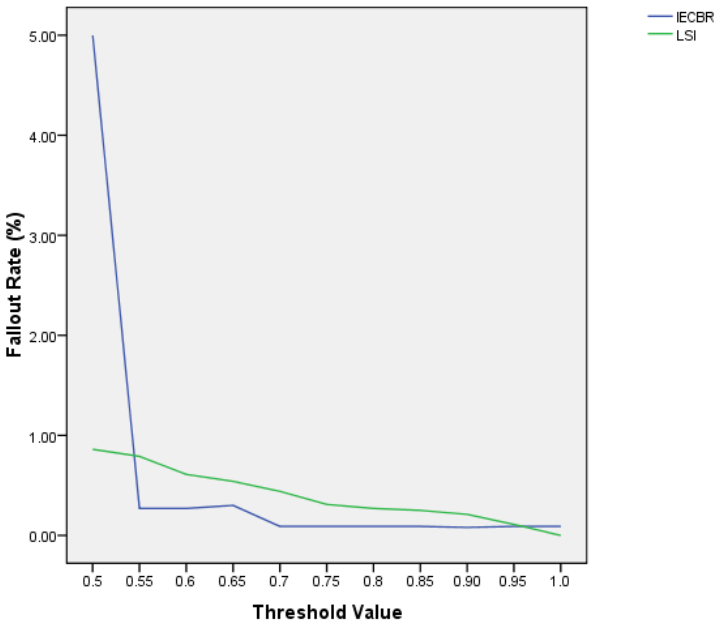

Fig. 10. Comparison of fallout rate between IECBR and LSI

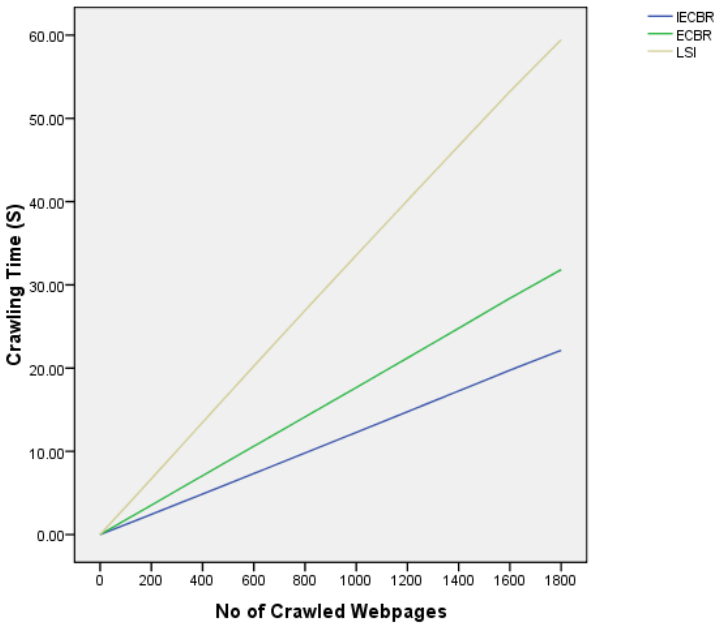

Fig. 11. Comparison of crawling time among IECBR, ECBR and LSI

Table 1.
Parameters of the testing data and the prototype

No. of crawled webpages	1800
Size of crawled webpages	49.8 megabytes
No. of generated HSDE metadata	1711
Size of generated HSDE metadata	3 megabytes
Size of the whole system	47.3 megabytes

Table 2.
The performance of IECBR on the optimal threshold value

Optimal Threshold Value	0.7
Harvest Rate	90.47%
Precision	88.19%
Mean Average Precision	88.53%
Recall	95.69%
F-Measure	91.56%
F-Measure(β =2)	94.09%
Fallout	0.09%