6S: Adding a Semantic Model To 5S Framework

Heba Neama, Yasser Fouad, Mohamed Kholeef
Alexandria University
Egypt
{heba_noima@yahoo.com} {y_fouad@alex.edu.gov} {kholeef@ast.edu}

ABSTRACT: *In this paper, we proposed new model for digital library which is an extension for the 5S Model to include the semantic web layer in the structure of digital library to be 6S model for digital library. We also discuss the important role of semantic web in digital library and how are semantic web technologies affect the information retrieval accuracy. We represent the semantic layer in 6S Model by adding ontology to a digital library that satisfies the 5S model and enhance ontology by updating it automatically. This ontology is used in books classification and retrieval according to concepts in ontology.*

Keywords: Semantic web, Digital library, Ontology, Hierarchical classification, Naive Bayes classifier

Received: 18 September 2015, Revised 25 October 2015, Accepted 4 November 2015

1. Introduction

Digital Libraries (DLs) are systems specifically designed to assist users in information seeking activities. As a result, libraries face new challenges, competitors, demands, and expectations [1]. Libraries are redesigning services and information products to add value to their services and to satisfy the changing information needs of the user community. Traditional libraries are still handling largely printed materials that are expensive and bulky. Information seekers are no longer satisfied with only printed materials. They want to supplement the printed information with more dynamic electronic resources [2]. In Section 2 there is brief description about semantic web. In section 3 there is brief definition about 5S Model of digital library. In section 4 we introduce the 6S proposed model for digital library. Section 5 is the ontology implementation and update ontology method. Section 6 is case study of applying algorithm to update ontology in digital library and Finally Section 7 is the Conclusion with future work.

2. Semantic Web

Semantic Web is a technology which adds well-defined documents on the Web for computers as well as people to understand the meaning of the documents more easily, and to automate the works such as information searches, interpretation, and integration. The ontologies, which are an essential component of the semantic Web, define the common words and concepts used to describe and represent an area of knowledge [3].

A semantic information search based on the ontology can provide the inferred and associated information between data [3]. The use of ontologies in the context of digital libraries could be interesting in order to incorporate new functionalities by

describing the relationships between elements. The concept of ontology introduced by the Semantic Web is a promising path to extend digital library formalisms with meaningful annotations [4].

The Semantic Web Stack, also known as Semantic Web Cake or Semantic Web Layer Cake, illustrates the architecture of the Semantic Web.

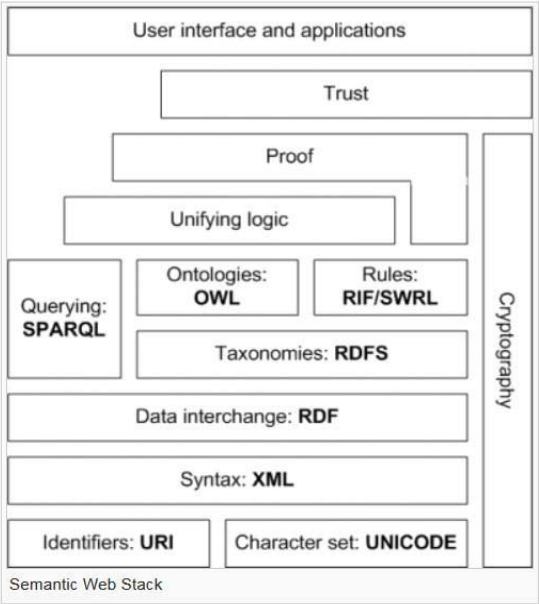

Figure 1. Semantic web stack

Each layer exploits and uses capabilities of the layers below. It shows how technologies that are standardized for Semantic Web are organized to make the Semantic Web possible [5].

3. 5S Formal Model for Digital Library

5S provides a formal framework to capture the complexities of DLs. The definitions in [6] unambiguously specify many key characteristics and behaviors of DLs. This also enables automatic mapping from 5S constructs to actual implementations as well as the study of qualitative properties of these constructs (e.g., completeness, consistency). In this section, we summarize the 5S theory from [6]. Here we take a minimalist approach, i.e., we describe briefly, according to our analysis, the minimum set of concepts required for a system to be considered a digital library [7].

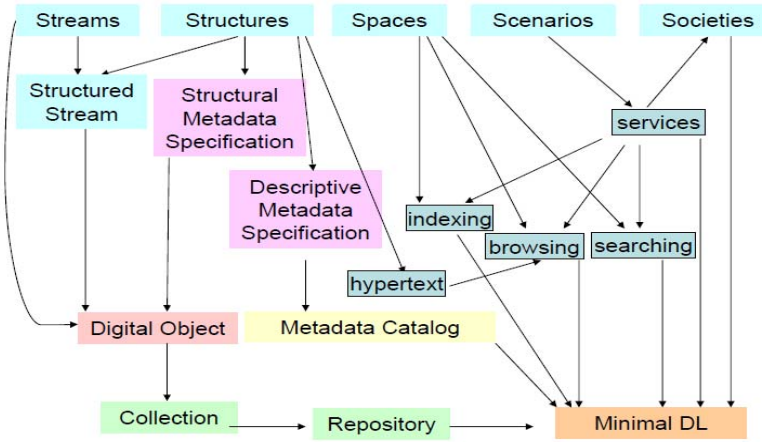

Figure 2. Digital library in 5S Framework

1- Streams are sequences of arbitrary types (e.g., bits, characters, pixels, frames) and may be static or dynamic (such as audio and video). Streams describe properties of DL content such as encoding and language for textual material or particular forms of multimedia data.

2- Structure specifies the way in which parts of a whole are arranged or organized. In DLs, structures can represent hypertexts, taxonomies, system connections, user relationships, and containment– to cite a few.

3- Space is a set of objects together with operations on those objects that obey certain constraints. Spaces define logical and presentational views of several DL components, and can be of type measurable, measure, probability, topological, metric, or vector space.

4- Scenario is a sequence of events that also can have a number of parameters. Events represent changes in computational states; parameters represent specific variables defining a state and their respective values. Scenarios detail the behavior of DL services.

5- Society is "*a set of entities and the relationships between them*" and can include both human users of a system as well as automatic software entities which have a certain role in system operation. These 5Ss, along with fundamental set theoretic definitions, are used to define other DL constructs such as digital objects, metadata specification, collection, repository, and services [7].

4. 6S Proposed Model for Digital Library Streams, Structures, Spaces, Scenarios, Societies and Semantic

Due to the important role of semantic web technologies in enhancing the digital library functionalities we propose new model which is an extension of the 5S formal model of digital library to include semantic web technologies.

Ontology can describe the different concepts of digital library entities and the relationship between those entities. Ontology enhances the digital library structure and functionalities with the meaning of classification, browsing and information retrieval in DL. In the following section we will describe the 6^{th} S in the 6S formal model.

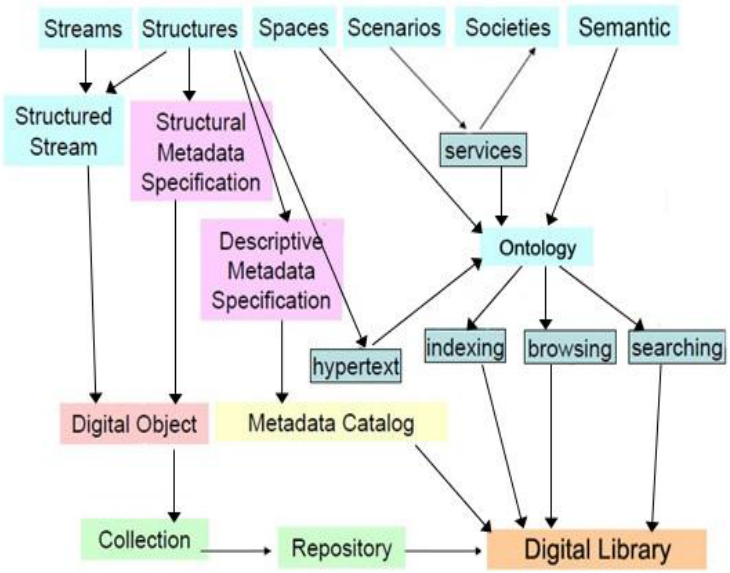

Figure 3. Digital library in 6S Framework

in The 6S proposed model for digital library in which we use ontology to define the concepts of digital library and the relationship between them to enhance the classification, browsing and information retrieval in digital library.

In figure 3 Semantic web technologies represents important layer where 5S layers depends on ontology to gain meaningful data from digital library while searching, browsing and indexing. Ontology must be updated automatically with every update in the digital library contents. In the following section we will discuss ontology based navigation and classification according to the 5S model definitions and how can we keep ontology up to date using hierarchal Algorithm.

Ss	Examples	Objectives
Streams	Text; video; audio; image	Describes properties of the DL content such as encoding and language for textual material or particular forms of multimedia data
Structures	Collection; catalog; hypertext; document; metadata	Specifies organizational aspects of the DL content (e.g., structured stream = DO or protocol), profiles, logs, P2P network, services
Spaces	Measure, measurable, topological, vector, probabilistic	Defines logical and presentational views of several DL components; host and user locations; GIS
Scenarios	Searching, browsing, recommending	Details the behavior of DL services, workflows, life cycle, preservation
Societies	Service managers, learners, teachers, etc.	Defines managers, responsible for running DL services; actors, that use those services; and relationships among them (including policies) 12
Semantic	Ontology represents digital library concepts and relationship between entities, Ontology based Information Retrieval.	Information quality, performance expectancy.

Table 1. 6S Model examples and objectives

4.1 Ontology-Based navigation and Classification

Ontologies specify relevant concepts, the types of things and their properties and the semantic relationships that exist between those concepts in a particular domain. Formal specifications use a language with a mathematically well-defined syntax and semantics to describe such concepts, properties, and relationships [8].

Definition 1: Field *fi* is a label associated with a node of a structural or descriptive metadata specification.

Definition 2: A query *q* is the representation of user interest or information need. The exact format of a query is left unspecified here since it is system-dependent.

Definition 3: $tfr \subset S_3 \times$ Spaces is a function that transforms any element of a concept in S_3 into a space. *Transformers* = $\{tfr_1, tfr_2, \ldots, tfr_n\}$ is a set of such functions.

Definition 4: Let $\{do_i\} = \{do_{i1}, do_{i2}, \ldots, do_{in}\}$ be a set of digital objects and *Ct* = {c1, c2,…, cm} be a set of labels for categories. A classifier $class_{Ct}:\{do_i\} \rightarrow 2^{Ct}$ is a function that maps a digital object to a set of categories.

Definition 5: A cluster $clu_k = \{do_{1k}, do_{2k}, \ldots, do_{nk}\}$ is a subset of a set of digital objects [8].

Service	User input	Other service Input	Output
Classifying	$\{do_i\}$	$class_{Ct}$, *Ct*	$\{(doi, \{c_k\_i\})\}$
Clustering	$\{do_i\}$	none	$\{clu_k\_i\}$
Searching	q, C_i	I_{C_i}	{dok}

Table 2. Services of DL

We can represent the Semantic in 6S digital library model as the procedure of applying semantic web technologies in digital library by defining and building ontology for digital library concepts and classifying data points according to these concepts.

5. Ontology Implementation

Here the Subject ontology in Figure 4 created in digital library is arranged in a tree hierarchy, at each level of the tree there are classes represent subjects of data points, to classify data point as a member of existing class we will start with the root node. Every data point belongs to the root.

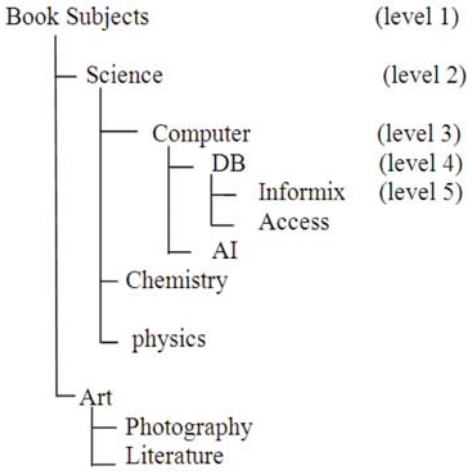

Figure 4. Subject ontology

5.1 Example of Ontology Constraints

Examples of class constraints are suggested by ontology as shown in figure 5 are as follows:

(1) The "*Subclass-Super class*" constraint: if a data point is member of "*Database*", then it should also be member of "*Computer*".

(2) The "*Mutual Exclusion*" constraint states that: if a data point belongs to "*Computer*" class, then it should not belong to "*Physics*" class

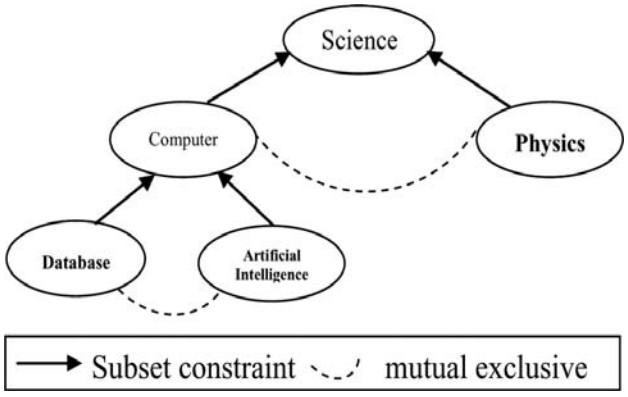

Figure 5. Constraints Model

5.2 Algorithm used for updating Ontology

The ontology in figure 4 can be updated using the hierarchical algorithm, which provides the ability to classify any data point to existing class or create new class and update the constraints according to the new added class.

We used the naïve bayes classifier formula to determine the probability distribution of data point over all classes, Naive bayes is able to compare not only single words, as in most current approaches, but substrings of an arbitrary length. We also used the Minmax statistic theory in algorithm as a criterion to take the decision of creating new class if needed.

5.2.1 Naïve Bayes

Naïve Bayes classifier is useful in our case study of the tree classifier. We begin with a set of training examples with each example document assigned to one of a fixed set of possible classes, $C = \{C_1, C_2, C_3,... C_J\}$. Naïve Bayes classifier uses this training data to generate a probabilistic model of each class; and then, given a new document to classify, it uses the class models and Bayes' rule to estimate the likelihood with which each class generated the new document. The document is then assigned to the most likely classes [10].

Hence, given a document $d = \{d_1, d_2, d_3,...,d_L\}$, we use Bayes theorem to estimate the probability of a class C_j:

$$P(C_j \mid d) = \frac{P(C_j)\, P(d \mid C_j)}{P(d)} \tag{1}$$

To combine multiple pieces of evidence it is that, if two different key words $w1$ and $w2$ the probability calculation will start with the following equation [10]:

$$P(C_1 \mid W_1 \wedge W_2) = \frac{P(C_1)\, P(W_1 \wedge W_2 \mid C_1)}{P(W_1 \wedge W_2)} \tag{2}$$

When two features are conditionally independent, we can calculate their co-occurrence as a simple multiplication [11]

$$P(C_1 \mid W_1 \wedge W_2) = \frac{P(W_1 \mid C_1)\, P(W_2 \mid C_1)\, P(C_1)}{P(W_1 \wedge W_2)} \tag{3}$$

Russell and Norvig explain that, we can eliminate the term $P(W_1 \wedge W_2)$ with normalization, which uses the conditional probabilities and the assumption of conditional independence to calculate this term[11].

$$P(C_1 \mid W_1 \wedge W_2) = \frac{P(W_1 \wedge W_2 \mid C_1)\, P(C_1)}{P(W_1 \wedge W_2)} \tag{4}$$

$$P(C_2 \mid W_1 \wedge W_2) = \frac{P(W_1 \wedge W_2 \mid C_2)\, P(C_2)}{P(W_1 \wedge W_2)} \tag{5}$$

The two equations sum to 1, since the word W_1 is certainly either related to C_1 or C_2 and then multiplies the whole equation by the common denominator and the resultant equation is

$$P(C_2 \mid W_1 \wedge W_2) = \frac{P(W_1 \mid C_1)\, P(W_2 \mid C_1)\, P(C_1)}{P(W_1 \mid C_1)\, P(W_2 \mid C_1)\, P(C_1) + P(W_1 \mid C_2)\, P(W_2 \mid C_2)\, P(C_2)} \tag{6}$$

5.2.2 Hierarchical Algorithm

Based on the ontology in figure 4, given a non-classified data point d and a set of classified data points D associated to classes C, each class has set of constraints $cons_i$, It is required to classify d in certain class whether by adding class or creating new class with its constrains $cons_{i+n}$.

Function Update_Ont_Algorithm (D., C, d, Consi):Cd, $Cons_{i+m}$,

Input:

D set of labeled data points,

C set of classes,

d unclassified data point,

$Cons_i$ set of constraints;

Output:

C_d class label of *d*,

$Cons_{i+m}$ constraints on new class;

P_{new} probability of creating a new class

h = Number of levels of ontology

for *J* = 1 to h do

for *K* = 1 to $|C_j|$ do

($|C_j|$ *is the number of classes in one level J*) *Using Naive Bayes classifiers Find* $P(C_{jk}|d)$

$$P(C_{jk}|d) = \frac{P(W_1|C_{jk})\, P(W_2|C_{jk})\, P(W_n|C_{jk})\, P(C_{jk})}{P(W_1|C_{jk})\, P(W_2|C_{jk}) P(W_n|C_{jk})\, P(C_{jk}) + P(W_1|C_{j(k+1)})\, P(W_2|C_{j(k+1)}) P(W_n|C_{j(k+1)})\, P(C_{j(k+1)})}$$

$$P(C_{j(k+1)})|d) = \frac{P(W_1|C_{j(k+1)})\, P(W_2|C_{j(k+1)}) P(W_n|C_{j(k+1)})\, P(C_{j(k+1)})}{P(W_1|C_{jk})\, P(W_2|C_{jk})\, P(W_n|C_{jk})\, P(C_{jk}) + P(W_1|C_{j(k+1)})\, P(W_2|C_{j(k+1)}) P(W_n|C_{j(k+1)}) P(C_{j(k+1)})}$$

C_d = **DataPointClassify** ($P(C_{jk}|d)$, $P(C_{j(k+1)}|\ d)$, *h*)

$Cons_i$ = **Update_Cons** ($\{d \cup D\}$,$\{Cd \cup I\}$, $Cons_i$)

end for

$Cons_{i+m} = Cons_i$

end for

end function

function DataPointClassify (*P*1, *P*2, *h*): C_d

Input:

*P*1 = $P(C_{jk}|\ d)$: probability of first class given a datapoint *d*,

*P*2 = $P(C_{j(k+1)}|\ d)$: probability of second class given a datapoint *d*

h: heigth of ontology]

Output:

C_d classification of d to certain class

for L = 2 to h **do**

if d has seed label at level L **then**

class (*d*, *levelL*) = *seed label*;

else

Classofdatapoint = *children*(*label*(*d*, $level_{l-1}$))

if Classofdatapoint is not empty **then**

if (*maxProb*(*P*1, *P*2) / *minProb*(*P*1, *P*2)) < 2

then Assign *d* to C_{dnew}

Set parent(C_{dnew}) = root class at level L-1

Else

C_d = *Class*(*maxProb* (*P*1, *P*2))

Assign d to maxProb(*P*1, *P*2)

end if

end if

end if

end for

end function

Function Update_Cons (*C*, *D*, $Cons^{old}$): $Cons^{new}$

Input:

***D*:** Datapoints;

***C*:** Class assignments all datapoints in D;

***$Cons^{old}$*:** Old constraints on the existing set of classes.

Output:

***$Cons^{new}$*:** Updated set of class constraints,

$Cons^{new} = Cons^{old} + C_{dNewCons}$;

Each new class created is added to a single parent at the time of creation.

Add each parent, child relationship as a constraint in $Cons^{new.}$

end function

6. Case Study

We created Subject Ontology of 5 levels (figure 4) hierarchy classes and super classes to associate subject of documents to concept in ontology. In this way user can view subjects organized in a classification scheme and can browse over this scheme to retrieve documents. This digital library is an open source Java project which is helping user to organize and retrieve documents in PDF format. It is software of Personal Information Management (PIM) that works with technologies of Semantic Web [12].

User can insert/edit information on documents like author, title, description, subject and so on. This information is stored as RDF (Resource Description Framework) and use standard properties like those defined in Dublin Core metadata set [12].

6.1 Digital library Stored Data
These are samples of data stored in digital library beside the other metadata of books.

The digital Library contains 200 PDF books with subject Database, 55 books related to access subclass of database, 65 books related to Informix subclass and 80 books of oracle subject.

In our case study we will compare between different methods of searching

1. Key word search without using semantic
Digital library can detect the subject of some PDF books from their metadata by using a powerful full-text search library written in Java. This results in low Recall = 5% and low precision = 25%

2. Searching using not updated ontology
Searching for certain books with subject is not exist in ontology will result in value zero of precision and recall because books are not classified according to any class or super class in hierarchical ontology

3. Searching after updating ontology
We need to retrieve book of certain subject and this subject is not exist in ontology so we will try to classify books to certain class in ontology or create new class according to book subject. In the following section we will apply algorithm to update ontology.

Labeled Data points	Classes
Book1: Introduction to Microsoft Access 2003 Book2: Microsoft-Access Tutorial Book3: Import Data into Microsoft Access	ACCESS
Book4: INFORMIX-4GL Reference Manual Book5: IBM Informix Administrator's Guide Book6: IBM Informix Storage Manager Administrator's Guide	INFORMIX
Book7: Oracle Automatic Storage Management Administrator's Guide	Unlabeled Data Points

Table 3. Digital library Sample data

6.2 Applying Hierarchical Algorithm
Starting from level 4 in subject ontology figure 4 assuming that Book7 from table3 is related to database super-class, we need to classify this unlabeled data point to one of the children which are (access and Informix) or creating new class in ontology for it.

Book7: Oracle Automatic Storage Management Administrator's Guide P(ACC) = 46% P(INF) = 54%

N1: number of books where the word exists in first class (access)

N2: number of books where the word exists in second class (Informix).

W/Access: number of access books where the word exists divided by the number of all books in access

W/Informix: number of Informix books where the word exists divided by the number of all Informix books.

Word	N1	N2	W/Acc	W/Inf
Oracle	20	25	0.36	0.39
Automatic	18	30	0.33	0.46
Storage	15	14	0.27	0.22
Management	27	20	0.49	0.3
Administrators	20	18	0.36	0.28
Guide	30	35	0.55	0.54

Table 4. Probability distribution over classes

7: Find $P(C_i|X)$ for all classes (Access and Informix)

$P(Access|X^u) = 0.6$

$P(Informix|X^u) = 0.4$

From the function of Consistent Assignment and since the Pcand of each class is nearly uniform then Create a new class C_{new}= "*Oracle*" at level L and assign the X^u to this new class

Set parent(C_{new}) = *class choice at level L*-1 (*Database*)

From the function of update constraints we update the constraints of the newly created class "*oracle*" as following:

1. Oracle is subclass of Database class (subset-constraints) and

2. Oracle class members cann't be member of any other class on the same level (mutually exclusive constraints)

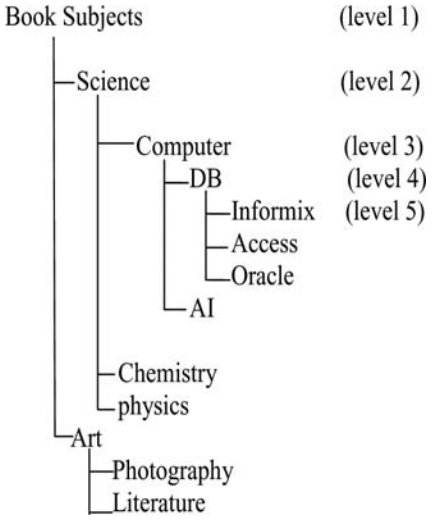

Figure 6. Updated ontology with new class Added

Searching After updating ontology shows enhancements in the values of precision and recall. From 80 books we retrieved 62 books relevant to subject and 10 irrelevant to it.

Precision = (62/62+10)*100 = 86.7%

Recall = (62/62+18)*100 = 77.5%

Updating ontology shows much better results of high precision and recall values than of keyword search and searching with incomplete ontology.

7. Conclusion and Future Work

Semantic Web technologies are valuable add ons for digital libraries. In this paper we proved that using ontology which is one of the semantic web technologies in digital library structure to define concepts and relationship between entities; it organize and gives meaningful metadata about digital library content and finally it improves information retrieval and books classifications in digital library. Key word searching was not effective with low precision and recall values. Searching in digital library using updated ontology results in the best values of precision and recall. The proposed Hierarchy algorithm is using the books keywords and using naïve bayes classifier to automatically classify books into the created subject ontology whether by creating new class or assign it to any existing class. Digital Library Structure should be modified to include the semantic web layer and this leads us to build new digital library model 6S as extension of the 5S model to use the semantic web technologies in digital libraries.

Future work will consist of evaluating the implementation and approach more carefully, validating the 6S digital library model with a number of quality aware case studies and using large digital library resources and different types of resources not only PDF files contents. Also future work should consist of measuring different semantic techniques with digital library to increase the quality of digital libraries.

References

[1] Suleman, H. (2002). Open digital libraries, in, Citeseer.

[2] Trivedi, M. (2010). Digital libraries: functionality, usability, and accessibility, *Library Philosophy and Practice* (e-journal), 381.

[3] Hwang, H. -S., Park, K. -S. , Kim, C. S. (2006). Ontology-based information search in the real world using web services, in: Computational Science and Its Applications-ICCSA, *Springer*, 125-133.

[4] Gómez-Berbís, J. M., Colomo-Palacios, R., García-Crespo, Á. (2008). CallimachusDL: using semantics to enhance search and retrieval in a digital library, *In*: Emerging Technologies and Information Systems for the Knowledge Society, *Springer*, 540-548.

[5] Berners-Lee, T., Hendler, J., Lassila, O. (2001). The semantic web, *Scientific American*, 284, 28-37.

[6] Gonçalves, M. A., Fox, E. A., Watson, L. T., Kipp, N. A. (2004). Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries, *ACM Transactions on Information Systems* (TOIS) 22, 270-312.

[7] Murthy, U., Gorton, D., Torres, R., Gonçalves, M., Fox, E., Delcambre, L. (2007). Extending the 5S digital library (DL) framework: From a minimal dl towards a dl reference model, *In*: Proceedings of the 1st Workshop on Digital Library Foundations, *ACM IEEE Joint Conference on Digital Libraries*, 25-30.

[8] Gonçalves, M. A., Fox, E. A., Watson, L.T. (2008). Towards a digital library theory: a formal digital library ontology, *International Journal on Digital Libraries*, 8, 91-114

[9] Dalvi, B., Cohen, W. W., Callan, J. (2013). Classifying entities into an incomplete ontology, *In:* Proceedings of the workshop on Automated knowledge base construction, ACM, 31-369.

[10] Pampapathi, R. M., Mirkin, B., Levene, M. (2005). A suffix tree approach to email filtering, arXiv preprint cs/0503030.

[11] Sinclair, S. (2004). Adapting Bayesian statistical spam filters to the server side, *Journal of Computing Sciences in Colleges*, 19, 344-346.

[12] Digital library Example URL: http://spdl.sourceforge.net/index.htm.