International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:11, 2011

# Organization Model of Semantic Document Repository and Search Techniques for Studying Information Technology

Nhon Do, Thuong Huynh, An Pham

***Abstract*—** Nowadays, organizing a repository of documents and resources for learning on a special field as Information Technology (IT), together with search techniques based on domain knowledge or document's content is an urgent need in practice of teaching, learning and researching. There have been several works related to methods of organization and search by content. However, the results are still limited and insufficient to meet user's demand for semantic document retrieval. This paper presents a solution for the organization of a repository that supports semantic representation and processing in search. The proposed solution is a model which integrates components such as an ontology describing domain knowledge, a database of document repository, semantic representation for documents and a file system; with problems, semantic processing techniques and advanced search techniques based on measuring semantic similarity. The solution is applied to build a IT learning materials management system of a university with semantic search function serving students, teachers, and manager as well. The application has been implemented, tested at the University of Information Technology, Ho Chi Minh City, Vietnam and has achieved good results.

***Keywords*—** document retrieval system, knowledge representation, document representation, semantic search, ontology.

## I. Introduction

THE electronic libraries and learning resource management systems are indispensable in the application of information technology in education and training. These systems are required to be increasingly effective, better serve learners, teachers and even managers to best satisfy their information need in teaching, learning and researching. This is one of the practical and urgent needs but the outcome is still very limited. The earlier works focused mostly on digitalizing documents and the application of information technology on library management, less focused on researching solutions for the management of learning resources on computer.

Current popular solutions and technologies have much support for the application of learning resource management, but mainly in the data processing. Some standards for resource description are proposed as LOM, IMS, Dublin Core, MARC ... but these standards are merely used to create metadata with simple description fields as title, license, author, ... and limited specific vocabulary, so not enough ability to interpret, combine resources by semantic content and thus features of the system is not sufficient to meet the increasing requirements, especially the organization, processing and integration of data, information and knowledge. For example, updating or searching documents is based not only on keywords (data), but also on the semantic content or related knowledge.

A document retrieval system (DRS) is a system finding documents in a database whose content is relevant to the information need of user. The current systems are largely based on the keywords and the popularity of the document. A list of keywords is a simple representation of content and shows the lowest level of information; and the semantic relationships between words (phrases) are not considered. The challenge for users of keyword based information retrieval systems is to describe information needs as a set of keywords and formulate a query to specify every possible form of a word that he believes may occur in the documents for which he is searching. Less experienced users can not specify the right keywords to search for their problems. These are the basic reasons why current search systems do not always return satisfied search results to users (ratio of number of useful documents retrieved on total number of documents retrieved is low; or cannot find the relevant documents when user provides synonymous keywords). These disadvantages caused difficulties for users in finding the exact information they need.

From the initial simple search model as Boolean, many authors have attempted to improve the efficiency of searching through the more complex models such as Vector Space Model [9], Probabilistic Models [4], and Language Model [7]. Many other works which have made effort to change weighting schemes, use natural language processing techniques [12,13], word sense disambiguation [8,10], query expansion [1] ... also contribute to increase search efficiency. Despite many proposals and efforts aimed at improving search results, the limitations of the use of keywords are not overcome yet.

Nowadays, in computer science there is a gradual shift to knowledge orientation or semantic processing. Accordingly, the concept based information retrieval systems have been researched and developed to replace the traditional systems which have revealed several major shortcomings. The search

Nhon Do, Department of Computer Science, University of Information Technology, Vietnam National University - Ho Chi Minh City, Vietnam, (email: nhondv@uit.edu.vn)
Thuong Huynh, Department of Mathematics and Computer Science, University of Science, Vietnam National University - Ho Chi Minh City, Vietnam (email: hh.thanhthuong@gmail.com)
An Pham, Department of Computer Science, University of Information Technology, Vietnam National University - Ho Chi Minh City, Vietnam, (email: pntruongan2005@gmail.com)

is based on space of concepts and semantic relationships between them. Semantic or conceptual approaches attempt to implement some degree of syntactic and semantic analysis; in other words, they try to reproduce to some degree of the understanding of the natural language text that a user would provide corresponding to what users think. In particular, the approach based on ontology is considered a modern approach and most appropriate for the representation and handling of content and meaning of documents [2, 5, 6, and 11]. In addition, many richer document representation schemes also proposed by considering not only words but also semantic relations between words as the semantic nets, conceptual graph, star graph, frequency graph, distance graph,... be evaluated with high potential because they allow to represent semantic links between concepts whereas poor representation models cannot.

The main goal of this paper is to introduce models, algorithms, and techniques for organizing text document repositories supporting representation, and dealing with semantic information in the search. The paper is organized as follows: section 2 introduces ontology model describing knowledge about a particular field as Information Technology; section 3 presents a graph based document representation model; section 4 introduces a model for organizing, storing document repository on computer; section 5 presents techniques in semantic search; finally a conclusion ends the paper.

## II. ONTOLOGY MODEL

Classed Keyphrase based Ontology model (CK_ONTO) is a system composed of six components:

$$(K, C, R_{KC}, R_{CC}, R_{KK}, \text{label})$$

, in which the components are described as follows:

- K is a set of keyphrases
- C is a set of classes of keyphrases
- $R_{KC}$ is a set of relations between keyphrase and class
- $R_{CC}$ is a set of relations between classes
- $R_{KK}$ is a set of relations between keyphrases
- label is labeling function for classifying keyphrase.

### *A. Set of keyphrases K*

Keyphrase is the main element to form the concept of ontology. In addition, keyphrase also means a structural linguistic unit as a word or a phrase. There are two kinds of keyphrases: single keyphrase and combined keyphrase. Single keyphrase only represents a concept, formed by a lexical item as a single word or a fixed phrase. For example, *computer*, *network*, *database*, *data structure*. Combined keyphrase represents several concepts, formed by a group of single keyphrases which have semantic relationships between components. For example, *computer networking and communication*, *computer graphics and image processing*, *database programming*, *network programming*.

Let K = {k | k is a keyphrase of knowledge domain}, $K = K_1 \cup K_2$, in which, $K_1$ is a set of single keyphrases and $K_2$ is a set of combined keyphrases.

### *B. Set of classes of keyphrases C*

Each class $c \in C$ is a set of keyphrases related to each other by a certain semantics. A keyphrase may belong to different classes. The classification of K depends on the specialization of concepts. Let $C = \{c \in \wp(K) \mid c \text{ is a class of keyphrases which describes the sub topics or sub subjects of knowledge domain}\}$. For example, *DATA STRUCTURE* class contains keyphrases related to data structures as follows: *DATA STRUCTURE = {stack, queue, contiguous list, linked list, hash table, graph, tree, sorting, strictly binary tree, complete binary tree, AVL tree, Red Black tree, Bubble sort, Merge sort, .... }*

### *C. Set of relations between keyphrase and class $R_{KC}$*

A binary relation between K and C is a subset of $K \times C$ and $R_{KC} = \{r \mid r \subseteq K \times C\}$. In this paper, $R_{KC}$ only includes a relation called "belongs to" between keyphrase and class, which is defined as a set of pairs (k, c) with $k \in K$, $c \in C$.

### *D. Set of relations between classes $R_{CC}$*

A binary relation on C is a subset of $C \times C$ and $R_{CC} = \{r \mid r \subseteq C \times C\}$. There are two types of relations between classes are considered:

- Hierarchical relation:

A class can include multiple sub classes or be included in other classes. A subclass is a class that inherits some properties from its superclass. The inheritance relationships of classes give rise to a hierarchy or a hierarchical relationship between classes. For instance, *Programming Language* and *Programming Technique* Are Subclasses Of *Programming*.

- Related relation:

According to the way to build a class above, a keyphrase may belong to many different classes or a subclass is allowed to have any number of father classes. This leads to the emergence of a relation on which the classes are called "related to each other" but not in meaning of inclusion or containment. These classes have some common properties, more or less related to each other because they have similar keyphrases or subclasses. For example, the related classes are *communication* and *network*, *hardware* and *electronic technology*.

### *E. Set of relations between keyphrases $R_{KK}$*

A binary relation on K is a subset of $K \times K$, i.e. a set of ordered pairs of keyphrases of K, and $R_{KK} = \{r \mid r \subseteq K \times K\}$ .There are several different kinds of semantic relations between keyphrases. The amount of relations may vary depending on considering the knowledge domain. These relations can be divided into three groups: equivalence relations, hierarchical relations, non-hierarchical relations.

Equivalence relations link keyphrases that have the same or similar meaning and can be used as alternatives for each other, such as synonym relation, abbreviation relation, near-synonym relation. For example, *JSP* is the short form of *Java Server*

*Page*, *Twittworking* is synonymous with *Twitter networking*, *semantic search* is close to *search by content.*

Hierarchical relations link keyphrases that one of which has a broader (more global) meaning than the other, such as "a part of" relation (or part-whole relation), "a kind of" relation (is-a relation).For example, *soft computing* is a part of *computer science*, *recognition* is a part of *image processing, semantic net* is a kind of *graph*, *Java* is a kind of *programming language*.

Non-hierarchical relations link keyphrases which are semantically related each other without forming a hierarchy or semantic equivalence, such as Expansion, Same-class, Cause, Influence, Instrument, Make, Possession, Source, Aim, Location, Temporal, Manner, Support, Beneficiary, Property, Agent, Circumstance, and Person.

*F. Labeling function for classifying keyphrase*

A keyphrase may refer to a terminology or a class to which the keyphrase belongs and its name is the same as name of the class. Thus, the semantics of a keyphrase may relate to its level of content (or level of its class) such as discipline, major, subject, theme, topic. To describe the information that a keyphrase represents a class and level of the class, a labeling function is used as follows:

Let Labels = {"discipline", "major", "subject", "theme", "topic", "terminology"} is a set of keyphrase labels. The function of labeling *label*: K → $\wp$ (Labels), in which each keyphrase is a "terminology" by default. For example, *grid computing* ↦ {"terminology", "major"}.

## III. Document Representation

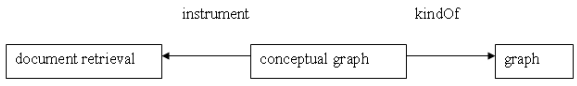
Understanding the document content involves not only the determination of the main keyphrases occur in that document but also the determination of semantic relations between these keyphrases. Therefore, each document can be represented by a graph of keyphrases in which keyphrases are connected to each other by semantic relations.

*Definition: A keyphrase graph (KG)* defined over a ontology CK_ONTO, is a triple ($G_K$, E, l) where:

- $G_K \subset$ K is the non-empty, finite set of keyphrases, called set of vertices of the graph.
- E is a finite set with elements in $G_K \times G_K$, called set of arcs of the graph. The arc is always directed and represents a semantic relation between its two adjacent vertices.
- $l : E \to R_{KK}$ is a labeling function for arcs. Every arc e ∈ E is labeled by relation name or relation symbol.

Keyphrase graph is a graph-based knowledge representation model. When these graphs are used for representing a document, keyphrase vertices represent keyphrases of CK_ONTO ontology treated in the document (reflect the main content or subject of the document), and the labeled arcs represent semantic links between these keyphrases. For example:

.




$G_K$ = { document retrieval, conceptual graph, graph}

E = {$e_1$ = ( conceptual graph, document retrieval), $e_2$ = (conceptual graph, graph)}

l($e_1$) = instrument, l($e_2$) = kindOf

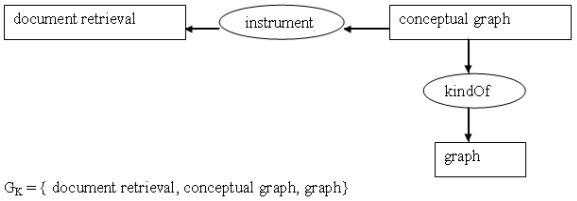
Fig. 1 An example of keyphrase graph

From the above definition of keyphrase graph G, we define an extensive keyphrase graph $G_e$ derived from G:

*Definition: An extensive keyphrase graph,* denoted as $G_e$, derived from keyphrase graph G = ($G_K$, E, l), is a triple ($G_K$, $G_R$, E') satisfying the following conditions:

- ($G_K$,$G_R$,E') is a bipartite, finite and directed graph,
- $G_K \subset$ K is a non-empty keyphrase vertex set.
- $G_R \subset R_{KK}$ is a relation vertex set which represents the semantic relations between keyphrases. (The vertex set of the graph is N =$G_K \cup G_R$, $G_K \cap G_R \neq \emptyset$). Each arc e ∈ E is correspond to a vertex $\tilde{r} \in G_R$ with $\tilde{r}$ = (e, lab(e))
- E' is a non-empty set with elements in $G_K \times G_R \cup G_R \times G_K$, called set of arcs of the graph. Vertices of the bipartite graph are divided into two nonempty, disjoint sets $G_K$ and $G_R$, with two different kinds of vertices. All arcs then connect exactly one vertex from $G_K$ and one vertex from $G_R$. Therefore, all arcs either go from a keyphrase vertex to a relation vertex or from a relation vertex to a keyphrase vertex.

This extensive keyphrase graph can be considered a variant of conceptual graph. There is 1:1 correspondence between a keyphrase graph and its extensive form. We can be easily transformed from the original keyphrase graph to the extensive graph and vice versa. Using which form of graph depends on its convenience in representation, storage, processing, calculation or implement. The illustration below is the same keyphrase graph in Figure 1 but in extensive form:




$G_K$ = { document retrieval, conceptual graph, graph}

$G_R$ = { $r_1$ = ( $e_1$, instrument), $r_2$ = ($e_2$, kindOf)}

E' = {e'$_1$ = ( conceptual graph, $r_1$), e'$_2$ = ($r_1$ , document retrieval), e'$_3$ = (conceptual graph, $r_2$), $e_4$ = ($r_2$, graph)}

Fig. 2 An extensive keyphrase graph

*Definition:* Let G = (K, R, E) be a keyphrase graph (in extensive form). A *sub keyphrase graph (subKG)* of G is a keyphrase graph G' = (K', R', E') such that:

$K' \subseteq K, R' \subseteq R, E' \subseteq E \text{ and } (i, j) \in E' \Rightarrow i, j \in K' \cup R'$.

A subKG of G can be obtained from G only by repeatedly deleting a relation vertex (and arcs incident to this vertex) or an isolated keyphrase vertex.

## IV. Semantic Document Base model

This section considers a model of organizing a document repository on computer that supports tasks such as accessing, processing and searching related to document content or the semantics. This model is called "Semantic Document Base" model (SDB model).

A SDB model is a system composed of five components, denoted by:

(D, FS, DB, CK_ONTO, SBD_R)

, in which the components are described as follows:

- D is a set of documents
- FS is a model of the file system of document repository
- DB is model of database of document repository
- CK_ONTO is an ontology describing domain knowledge
- SDB_R is a set of relationships between components.

### *A. Set of documents D*

This is a collection of real document not classified or handled. Each document d ∈ D has physical representation in the storage system as a file. However, in practice there are many documents stored in some files, i.e. that each document can include several sections each of which is stored into a separate file, but in scope of this research, each document is considered as a file.

### *B. Model of file system of document repository FS*

Storage system is organized according to hierarchical system of directories, or hierarchical directory tree. A directory can contain many subdirectories or documents, whereas each directory or document can only have one parent directory. A distinctive feature of the FS is that naming directories and organizing the directory hierarchy as well as classifying documents into directories must follow some predefined rules, in which the rules are described as follows:

Directory Naming rule: directory name must be normalized by the keyphrase representing a certain class defined in the CK_ONTO. That is, each directory corresponds to a class in the ontology describing the sub topic in the knowledge domain.

Hierarchical Organization rule: The hierarchy between directories must follow the hierarchy of classes in the CK_ONTO. For example, directory *Automatic Control* is a subdirectory of *Computer Engineering* corresponding to the hierarchical relationship between classes *automatic control* and *computer engineering*. For the directory system of learning documents, the hierarchy is made from the wide range such as discipline and major to the narrower range such as courses, subjects or topics.

Rule of classifying documents into directories: Each document is represented by a list of keyphrases that describe major topics of the document, and each directory is also named by a keyphrase expressing semantic information. Then, measuring the semantic similarity between keyphrases representing directories and keyphrases representing a document give a way of classifying the document into a corresponding directory.

### *C. Model of database of document repository DB*

The database of document repository is created based on the relational database model and Dublin Core standards. Besides the common elements of Dublin Core, each document includes some own special features and attributes to express its information structure in more detail. For example, the information structure of the thesis includes own features such as scientific advisors, thesis defense committee and marks.

### *D. An ontology describing domain knowledge CK_ONTO*

The ontology model describes knowledge of the domain (as presented in Section 2) is a knowledge representation model for a special domain, including six components: (1) the set K of keyphrases, (2) the set C of classes of keyphrases describing sub subjects in the knowledge domain, (3) the set $R_{KC}$ of relations between the keyphrase and class, (4) the set $R_{CC}$ of relations between classes, (5) the set $R_{KK}$ of relations between the keyphrases, and finally a labeling function used for classifying keyphrase based on its level of content.

### *E. Set of relationships between components SDB_R*

All relationships between the components in the SDB model called Semantic Document Base – Relationship (SDB-R) includes:

1/. Each document d ∈ D is stored in a unique directory of the FS system, that determines a mapping:

$$pos : D \to FS$$
$$d \mapsto pos(d)$$

, for each document d ∈ D, there is a path pos(d) referring to a node on the FS directory tree.

2 /. Each document d ∈ D has a record in the database DB.

$$record : D \to r(DOCUMENT) \in DB$$
$$d \mapsto record(d) = t$$

Each tuple t of the relation r (DOCUMENT) stores information of a real document d with title, author, keywords, description, the name of the physical file,…and the attribute idDocument is used as the primary key to distinguish one document from another.

3/ Each document d ∈ D is represented by a keyphrase graph KD (d) ∈ $F_{KG}$ ($F_{KG}$ is a set of keyphrase graphs) in which keyphrase vertices represent keyphrases of CK_ONTO treated in the document and relation vertices represent semantic relations between these keyphrases.

$$KG : D \to F_{KG}$$
$$d \mapsto KG(d)$$

4/ Each directory in FS corresponds to a class in ontology CK_ONTO and the hierarchical relation between directories depends on the hierarchical relation between classes of the ontology. Then, there is a mapping:

$$cl : X \to C$$
$$x \mapsto cl(x)$$

so that for all x, y ∈ X, if x f y then cl(y) ⊂ cl(x)

, in which X is the set of directory names and f is the hierarchical relation.

The relationships of components in the SDB model is illustrated in the following diagram

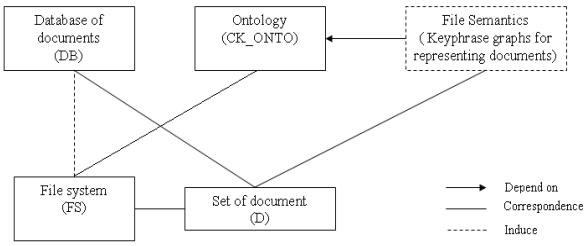



Fig.3 Relationship between the components in the SDB model

## V. Semantic search

This section will discuss an approach for semantic search based on relevance evaluation between the target query and documents by calculating measures of semantic similarity between keyphrases, relations and keyphrase graphs representing documents. The definitions of semantic similarities are given based on ideas of D. Gennest and M. Chein [3] with some modifications.

### *A. Relevance evaluation*

A keyphrase graph is constituted by keyphrases and relations, so the direction to measure semantic similarity between graphs is to calculate the similarity between keyphrases and the similarity between relations used in the graphs.

Let $\alpha : K \times K \to [0,1]$ and $\beta : R_{KK} \times R_{KK} \to [0,1]$ be two mappings to measure semantic similarity between two keyphrases and two relations defined in the CK_ONTO ontology. 1 represents the equivalence between two objects and 0 corresponds to the lack of any semantic link between them. The values of β are selected manually based on the opinions of experts of the field. Determining manually the values of β is possible because of the small number of relations.

Definition: Let k, k' ∈ K, a binary relation P on K defined as : P (k, k') iff k = k' or ∃ $S = (s_1, s_2, ..., s_n)$ a sequence of integers ∈ [1, t] ( t = $|R_{KK}|$) such that $k\, r_{s_1}\, k_1,\ k_1\, r_{s_2}\, k_2,\ ...,\ k_{n-1}\, r_{s_n}\, k'$ with $r_i$ is a relation of $R_{KK}$ (for all x and y in K, x has a relation r with y if and only if (x, y) ∈ r, written as x r y ).

The mapping α may be defined by using the sequence used in the relation P as follows:

$$\alpha(k,k') = 0 \ \text{if not} P(k,k')$$

$$\alpha(k,k') = Max\{V(k\, r_{s_1}\, k_1,\ k_1\, r_{s_2}\, k_2,\ ...,\ k_{n-1}\, r_{s_n}\, k')\} \quad (1)$$

if

∃ $S = (s_1, s_2, ..., s_n)$ a sequence of integers ∈ [1, t] ( t = $|R_{KK}|$) such that $k\, r_{s_1}\, k_1,\ k_1\, r_{s_2}\, k_2,\ ...,\ k_{n-1}\, r_{s_n}\, k'$.

Mapping V allows to consider the various semantic relations used in the sequence, is defined as:

$$V(k\, r_{s_1}\, k_1,\ k_1\, r_{s_2}\, k_2,\ ...,\ k_{n-1}\, r_{s_n}\, k') = \prod_{1}^{n} val\_r_{s_i}(k_{i-1}, k_i)\ \ (k_n \equiv k') \quad (2)$$

, in which, $val\_r_{s_i}(k_{i-1}, k_i)$ is the weight assigned to relations $r_{s_i}$ over pair of keyphrases ($k_{i-1}$, $k_i$). This weight is a measure of semantic similarity between the keyphrases $k_{i-1}$ and $k_i$ which are directly linked by the relation $r_{s_i}$. The value of $val\_r_{s_i}(k_{i-1}, k_i)$ is determined using expert method.

The mapping V allows to evaluate the combination of semantic relations used in sequence. This is necessary because the semantic similarity between two keyphrases linked by a semantic relation may vary depending on the used relation. Some links represent a large difference in meaning while other links represent small semantic distance. For example, keyphrases linked by a synonym relation are more semantically likeness than keyphrases linked by a hierarchical relation. Moreover, pairs of keyphases linked by the same relation may have different semantic similarity. For instance, in a hierarchy tree, the links closer to the root node often have greater semantic distance than the lower-level links. If there may exists many sequences from k to k', value of α(k, k') depends on the maximum of V.

*Definition:* Let H = (KH, RH, EH) and G = (KG, RG, EG) be two keyphrase graphs defined over CK_ONTO. A *projection* from H to G is an ordered pair $\Pi = (f, g)$ of two mappings $f : RH \to RG, g : KH \to KG$ satisfying the following conditions:

- Projection preserves the relationships between vertices and arcs, i.e. for all $r \in RH, g(adj_i(r)) = adj_i(f(r))$ , $adj_i(r)$ denotes the $i^{th}$ vertex adjacent to relation vertex r.
- $r \in RH, \beta(r, f(r)) \neq 0$
- $k \in KH, \alpha(k, g(k)) \neq 0$

*Definition:* A valuation pattern of a projection $\Pi = (f, g)$ from a keyphrase graph H to a keyphrase graph G is defined as follows:

$$v(\Pi) = \frac{\sum_{k \in KH} \alpha(k, g(k)) + \sum_{r \in RH} \beta(r, f(r))}{|KH| + |RH|} \quad (3)$$

*Definition:* There is a partial projection from a keyphrase graph H to a keyphrase graph G iff there exists a projection from H', a sub keyphrase graph (subKG) of H, to G.

A valuation pattern of partial projection $v(\Pi_{partial})$ only depends on vertices of H' and is defined like projection $v(\Pi)$.

Based on valuation of projections, semantic similarity between two keyphrase graphs calculus is defined as:

$$Rel(H, G) = Max\{v(\Pi) \mid \Pi \text{ is partial projection from H to G}\} \quad (4)$$

Determining if a document is relevant for a user query and estimate this relevance is done by calculating the semantic similarity between the keyphrase graphs that represent them. Figure 4 shows the indexation of document called #14 and the best projection from the query with relevance ratio 86%

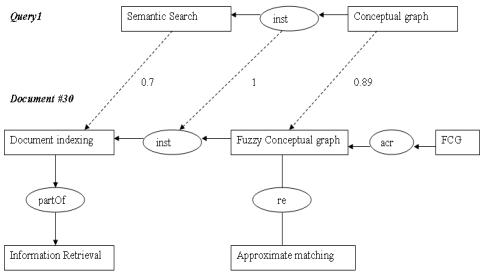


Fig. 4 Matching keyphrase graph

*Algorithm for calculating Rel(H,G) is described as following:*

Input: two keyphrase graphs H, G (both in extensive form)
Output: Rel(H,G)
Step 1: Variable Initialization
Sub_KG :={} // set of all H's sub keyphrase graphs
Projection :={} // set of all partial projections from H to G
Value :={} // values of each projection in Projection
Step 2: Find all sub keyphrase graphs of H
Sub_KG ← Find_SubKG(H);
Step 3: Find all projections from sub-keyphrase graphs of H to G
for kg in Sub_KG do
// Find all projections from kg to G and assign to Projection
Projection ← Projection ∪ Find_Projection(kg, G)
Step 4: Calculate the value for each projection in Projection
Value ← $v(\Pi)$
Step 5: Find $Rel(H,G) = Max\{v(\Pi)\}$

*B. Semantic search algorithm*

Semantic search algorithm is described as following:
Input:
- Document repository organized according to SDB model. Documents of SDB are represented by a set of keyphrase graphs KG(D) = {$G_1$, $G_2$, …, $G_k$}.
- User's query q.

Output: A list of ranked documents that relevant to the query q.
Step 1: Analysis and represent the query q by a keyphrase graph KG(q).
Step 2: Look for documents corresponding to the query
<2.1> Search in KG(G) keyphrase graphs which match with KG(q)
for g in KG(D)
if match(g, KG(q)) then
Result ← (g, Rel(g, KG(q))
<2.2> Rank documents in Result by the corresponding Rel value of each element.
Step 3: Display search results and suggestions for adjusting the query.
Step 4: Modify the query and repeat from step 1 until user is satisfied.

## VI. Conclusion

A solution for the organization of a semantic document repository that supports semantic representation and processing in search is described. The proposed solution is a model which integrates components such as an ontology of the relevant domain, a database, semantic representation for documents and a file system; with semantic processing and searching techniques. The solution is applied to build a IT learning materials management system of a university with semantic or document content based search function. The application has been implemented, tested at the University of Information Technology Ho Chi Minh City, Vietnam and search results have been highly appreciated by users. The research results will be the basis and tools for building many resource management systems in various different fields.

## References

[1] Aly, A.A, "Using a query expansion technique to improve document retrieval", International Journal "Information Technologies and Knowledge" (2008).
[2] Dario Bonino, Fulvio Corno, Laura Farinetti, Alessio Bosca , "Ontology Driven Semantic Search", WSEAS Transaction on Information Science and Application, Issue 6, Volume 1, pp. 1597-1605 (2004).
[3] D. Genest, M. Chein, "An experiment in Document Retrieval using Conceptual Graph" , Proceeding of 5th ICCS Conference, Washington, USA, p 489-504 (1997).
[4] Harter, S.P., "A probabilistic approach to automatic keyword indexing", PhD thesis, Graduate Library, The University of Chicago, Thesis No. T25146.
[5] Henrik Bulskov Styltsvig, "Ontology-based Information Retrieval", A dissertation Presented to the Faculties of Roskilde University in Partial Fulfillment of the Requirement for the Degree of Doctor of Philosophy (2006).
[6] Henrik Eriksso, "The semantic-document approach to combining documents and ontologies", International Journal of Human-Computer Studies Volume 65, Issue 7, Pages 624-639 (2007)
[7] Kraaij, W., "Variations on Language Modeling for Information Retrieval", ACM SIGIR Forum (2005).
[8] Sanderson M., "Word Sense Disambiguation and Information Retrieval", Annual ACM Conference on Research and Development in Information Retrieval, Ireland Springer-Verlag New York, Inc (1994)
[9] Salton G., A. Wong, and C.S. Yang, "A Vector Space Model for Automatic Indexing", Communications of the ACM, 1975. 18(11): p. 613–620.
[10] Stokoe, C., M.P. Oakes, and J. Tait, "Word sense disambiguation in information retrieval revisited", Annual ACM Conference on Research and Development in Information Retrieval Toronto, Canada (2003).
[11] Thanh Tran, Philipp Cimiano, Sebastian Rudolph and Rudi Studer, "Ontology-Based Interpretation of Keywords for Semantic Search", The Semantic Web Lecture Notes in Computer Science, Volume 4825/2007, 523-536 (2007)
[12] Tzoukermann, E., J.L. Klavans, and C. Jacquemin, "Effective use of natural language processing techniques for automatic conflation of multi-word terms: the role of derivational morphology, part of speech tagging, and shallow parsing", SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, p. 148–155 (1997)
[13] Vallez, M. and R. Pedraza-Jimenez, "Natural Language Processing in Textual Information Retrieval and Related Topics", I.S.S.o.t.P.F. University (2007).