Improved Semantic Representation and Search Techniques in a Document Retrieval System Design

Nhon V. Do, TruongAn PhamNguyen, Hung K. Chau, and ThanhThuong T. Huynh
University of Information Technology, Vietnam National University, HoChiMinh City
Email: {nhondv, truonganpn, hungck, thuonghtt}@ uit.edu.vn

***Abstract*—Recently, there has been a growing concern on processing documents'content and meaning in information retrieval; concept-based systems have been being studied and developed in order to replace the traditional ones that have several existing major weaknesses. We proposed a document retrieval system design in a specific domain, which manages semantic information related to document content and supports semantic representation and processing in document retrieval, and successfully applied it to some real life projects. However, the solution still has some limitations thus can be further developed and can be adapted for future requirements such as expending domain knowledge and range of applications, improving search results and processing speed. This paper presents some improvements in ontology model along with semantic processing techniques. These changes have been implemented in the same project with a previous solution to evaluate the effectiveness of this work.**

***Index Terms*—Document retrieval system, semantic representation and search, document representation, ontology**

I. Introduction

Nowadays, the need to seek valuable information in the enormous amount of available information is becoming more and more critical. Especially in scientific and academic community, literature research plays an undeniably important role, therefore, learning and researching material retrieval is an obvious and practical demand. Electronic libraries and learning resource management systems are indispensable to serve users better in teaching, learning and researching. These systems are required to be increasingly effective but their ability is still very limited.

The concept based information retrieval systems are being researched and developed to replace traditional systems that have revealed several major shortcomings. A better performance is expected from a retrieval system that considers semantic aspects, in which the search is based on a space of concepts and semantic relationships between them. Semantic or conceptual approaches attempt to make computers capable of understanding the meaning of words, phrases or natural language texts that users provide corresponding to what they think. One of the new approaches for semantic search that have received increasing interest recently is based on exploiting the ontology combined with techniques in machine learning and natural language processing. Ontology design has become an active research area in artificial intelligence in recent years. Application of ontologies in a specific domain has been discussed by many researchers including [2] [3] [5]. In fact, the approach based on ontology is considered as a modern approach and most appropriate for representing and processing the content and meaning of the document.

Recent popular approach in knowledge-based applications is the combination of linguistic ontologies and structured semantic matching. Semantic matching is one of the promising ways to improve both recall and precision of information retrieval. In [7]-[9], the authors proposed matching Conceptual Graphs that describe documents' contents for semantic search. The measurement of concept similarity or semantic distance between concepts was also discussed in [1] [4]. In [6], we adopted their original idea and made some modifications to suit to our work. One of the important contributions described in that paper is relevance evaluation between a query and documents by calculating measures of semantic similarity between keyphrases, relations and keyphrase graphs representing documents based on an existing ontology of the relevant domain.

II. A Design for Semantic Document Retrieval System

In [6], we have proposed a solution for the organization of a semantic document repository in a specific domain, which manages semantic information related to document content and supports semantic representation and processing in document retrieval. The solution includes a model called SDB (Semantic Base Document) with problems, semantic processing techniques and advanced search techniques based on measuring semantic similarity. Theoretically, we have contributed to the development of several models that can be used to design document retrieval systems in many different knowledge domains. These models include:

Manuscript received December 20, 2014; revised August 14, 2015.

doi: 10.12720/jait.6.3.146-150

An ontology model (called CK_ONTO) describes the knowledge in a particular field, in which keyphrases are used as the main element to form the concepts of ontology. The structure of the ontology is general and can be easily extended to many different knowledge domains as well as the different types of applications. The model includes six main components: (1) a set of keyphrases represents concepts in the domain, (2) a set of classes describes the topics of the domain, each class is a set of keyphrases related to each other in certain semantic sense, (3) a set of relations between the keyphrase and class, (4) a set of relations between classes, (5) a set of relations between keyphrases and finally (6) as a labeling function for classifying keyphrases.

A Keyphrase Graph model and its extensive form are used for the semantic representation of documents and defined over ontology CK_ONTO.

The SDB model, a model for organizing and managing document repository on computer that supports tasks such as accessing, processing and searching based on document content and meaning. This model integrates components such as: (1) a collection of documents, each document has a file in the storage system, (2) a file storage system with the rules on naming directories, organizing the directory hierarchy and classifying documents into directories, (3) a database of collected documents based on the relational database model and Dublin Core standard (besides the common Dublin Core elements, each document may include some special attributes and semantic features related to its content) , (4) an ontology describes partial knowledge of the relevant domain and finally (5) a set of relations between these components. The solution aims to build some document retrieval systems with main tasks including but not limited to organization, storage, searching and retrieval of text document, especially the ability to semantic search based on documents' content. At first, it was applied to build the learning resource repository management system [6], implemented and tested at the University of Information Technology HCM City, Vietnam. The initial experimental results show that the proposed solution is positive, effective and has good usability. Later, the solution was also applied to build the Vietnamese online news aggregating system supporting semantic processing for newspaper article in Labor & Employment and Public Investment & Foreign Investment domain. The system was implemented and tested at Binh Duong Department of Information and Communications, Binh Duong province, Viet Nam, and achieved impressive results.

These promising results fundamentally proved that the solution would be the basis for building many resource management systems in various different fields. However, the solution still has several shortcomings thus can be further improved and can be adjusted to new requirements in future such as expanding knowledge domain and range of applications, improving search accuracy and processing speed. To achieve that, ameliorating the ontology and semantic search techniques are considered top priority necessity for the current solution.

III. ONTOLOGY MODEL

This section, we present an ontology model which is revised from the old model mentioned in section 2.

A. The Components of the Mode

Advanced Classed Keyphrase based Ontology model is a system composed of five components: (K, C, R, Rules, label).

K is set of keyphrases. A keyphrase is a structural linguistic unit such as a word or a phrase. It's the main element to form the concepts of ontology. There aretwo kinds of keyphrases: single keyphrase and combined keyphrase.

C is set of classes. A class in C is a system consisting of three components: (K_b, Attr, Inst). In which, $K_b \subset K$ is a set of base keyphrases, Attr is a set of attributes and Inst is a set of instances. Base keyphrases are keyphrases playing a semantically important part in the definition, state in natural language, of a concept. The name of the concept is also the name of a class. An attribute (a property, a slot, etc.) of a class describes its interior structure. We only consider attributes which are keyphrases or Instance-type slots. An instance of a class is a specific object. Instances represent elements or individuals in an ontology. The name of an instance is the name of a keyphrase $k \subset K$ and the structure of instances depends on the structure of classes.

Set of relations in knowledge domain R consisting of three sub-sets: (R_{CC}, R_{KC}, R_{KK}).

R_{CC} is a set of relations between classes. A binary relation on C is a subset of $C \times C$. In this paper, R_{CC} includes three relations: Hierarchical relation, "A part of" relation and related relation. The class hierarchy represents a hierarchical relation (also called an "is-a" relation): a class A is a sub class of B if A inherits some properties from B, its superclass, and every instance of A is also an instance of B. "A part of" relations represent the relationship between the components of class and class in meaning of inclusion or containment. Class A has "a part of" relationship with class B if A represents a property of B. Related relations represent the semantic relationships between the elements of one class and one other class. Class A has related the relationship with class B if there is a class C with which A has "a part of" relationship, and Chas "a part of" relationship with B.

R_{KC} is a set of relations between keyphrase and class. A binary relation between K and C is a subset of $K \times C$. There are two kinds of relations between keyphrase and class: "A part of" and related relations. Similar to "a part of" relations and related relations on C, "a part of" relations between K and C represent the relationships between the components of class and class in meaning of inclusion or containment, and related relations between K and C represent the connections between the elements of one class and one other class.

R_{KK} is a set of relations between keyphrases. A binary relation on K is a subset of $K \times K$, i.e. a set of ordered pairs of keyphrases of 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 and non-hierarchical relations.

Rules is a set of deductive rules on facts related to keyphrases, classes or the property of relations. Each rule has the structure r: {h_1, h_2,..., h_n} → { g_1, g_2,..., g_m} with h_1, h_2,..., h_n are hypothesis facts and g_1, g_2,..., g_m are goal facts of the rule. In this model, there are two types of facts:

- Fact of kind 1: information about the property of relations, expressed with the structure: [<relation>, <property_of_relation>]. eg: [R_{sym}, "symmetric"].
- Fact of kind 2: the relationship between two objects: [<$object_1$>, <relation>, <$object_2$>]. eg: [k_1, R_{syn}, k_2], [k, R_{part_of}, c].

With that structure some examples of rule can be defined like rule of symmetric relation r_{sym}: {[R_1, "symmetric"], [k_1, R_1, k_2]} → {[k_2, R_1, k_1]}; rule of transitivity of relation $r_{transit}$: {[k_1, R_{syn}, k_2], [k_2, R_{kind_of}, k_3]} → {[k_1, R_{kind_of}, k_3]

***Labelling function for classifying keyphrases**:* 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. For example, soft computing {"terminology", "major"}.

B. Differences between the Two Ontology Models

With the revised ontology model, the system has kept all advantages from the old model and had some improvements. Class is a fundamental element to present the knowledge of a domain. Therefore, defining the set of classes well will help an ontology showing fully the information semantics. Because component C in the old model that classifies keyphrases is very simply; the ability of presenting semantics is still low, it has been revised as above. As a result, the current system has the capability of processing more complicated queries thanks to the new C. Moreover, the structure of a class contains a lot of information of a concept, so it will be exploited more in future.

The set of relations has been expanded with more relations. The set of deductive rules added in the ontology enables the system to determine the semantic relationship between two objects automatically. This is a remarkable improvement in calculating the relevance between two keyphrase graphs in general, in measuring the semantic similarity between two keyphrases.

The component label in CK_ONTO model was defined without using. Transforming the component C from CK_ONTO model to the advanced one by creating relations and labeling keyphrases has exploited this component in the system. This has proved that the fact we defined label element in the ontology model is appropriate and useful.

C. Automatic Semantic Relation Inference

Give a set of semantic relations on keyphrases and two different keyphrases. The inference engine will infer to find a certain semantic relation between these two keyphrases from the relation set in the ontology, the property of the relations and the set of deductive rules which have been defined already. In old solution [6], to calculate the relevance between two keyphrases k and k' (α), the system must find out sequences that connect k and k'. This method costs numerous of time because of exhausting all the whole relations. Moreover, in some cases, k and k' in reality do not have any relationship but linking sequences of them still exists in the system. On the contrary, with the new set of rules and forward chaining reasoning, measuring the similarity between two keyphrases is now more precise and faster.

IV. SEMANTIC SEARCH

This section discusses the approach to semantic search based on the evaluation of relevance, or semantic similarity between query and documents.

A. Weighting in Documents' Keyphrase Graph

The representation power of keyphrase graph can be vastly improved by assigning weights to its keyphrase vertices. In our previous work, we proposed two weighting value for document's keyphrase graph. The "term frequency" reflect a keyphrase's importance according to the number of times it appears in document, and the "importance of Position" (ip) determines the importance according to where it appears. However the formula we chose back then to calculate those weighting frequently yield a value too small. The small weighting value result in small similarity evaluation value, thus making search result ranking harder and may as well impair search precision.

So after testing and reconsideration, we have revised the formula for keyphrase weighting. The "term frequency" (tf) of keyphrase *k* in the document *d* will be defined as follows:

$$tf(k,d) = c + (1-c)\frac{n_{k,d}}{\max(n_{k',d} \mid k' \in d)} \quad (1)$$

where $n_{k,d}$ is the number of occurrences of keyphase *k* in document *d*. Parameter $c \in [0, 1]$ is the predefined minimum tf value for every keyphrases. The value of *c* is chosen through experimenting. This modification make sure no keyphrase will received a too small tf, which make sense since in our previous keyphrase was extraction in supervised manner. The new formula also possess more flexibility, the value of parameter *c* can be tuned to suit specific application.

$$ip(k,d) = a + (1-a)\frac{\sum_{i \in A} w_i}{\sum_i w_i} \quad (2)$$

In which w_i is the weight assigned for the i^{th}component of document *d*, *i* is the index of that component and the set of the index of all components in which k appear defined as$A=\{x/n_x\ (k,d)>0\}$. Parameter $a=max(w_i \mid i \in \in A)$ is the weight of the most important component where *k* appears, also serves as the predefined minimum value for ip(k, d.). The number of a document's component and the weight for each component is different for each type of document. A

paper, for example, contains title, abstract, keyword, main content and reference, with title and abstract often have the largest weight. With this new formula, we correct a problem in previous work and ensure that a keyphrase which appears in title as well as abstract will have higher ip than a keyphrase appear *only* in title.

B. Weighting in Query's Keyphrase Graph

A user query will be interpreted as a list of keyphrases and represented as a keyphrase graph. However, unlike a document, query lacks structure and context semantic, the weighting on query's keyphrase graph is, therefore, more complicated. To set the grounds for the weighting process, some principles must be established to guesstimate the importance of keyphrases.

If two keyphrases are part of a combined keyphrase, the structural information of that combined keyphrase can be looked up in the ontology to figure out which component is the main keyphrase. If such information is not available, we assume that the keyphase goes first in writing order is grammatically more important because it tell us more information about the second keyphrase. When two keyphrases are not part of a combined keyphrase, we can estimate their importance based on their position in the hierarchical relations tree. The keyphrase that goes deeper in the tree would be more important since it has more specific meaning and can help us process the query with more precise.

For the weighting of query's keyphrase graph, a strategy was chosen so that the sum of all the keyphrase's weight equal to 1. First, we apply the two principles above to every pairs of keyphrase in the query to decide which one is more important. Then a graph can be plotted based on this "more important" relation. If keyphrase *a* is more important than keyphrase *b*, we assume there is a "link" from *b* to *a*, that is to say "*b* link to *a*" or "*a* is linked to by *b*". The famous PageRank algorithm will then be used to weight keyphrase.

C. Semantic Relevance Evaluation

To calculating the relevance between keyphrase graphs, we inherit the basic method from our previous work [7] but with a revise formula to evaluate a projection between two keyphrase graphs:

$$v_w(\Pi) = \frac{|KH| * \sum_{k \in KH} tf(g(k),G) * \alpha(k,g(k)) * ip(g(k),G) * W(k) + \sum_{r \in RH} \beta(r, f(r))}{|KH| + |RH|} \quad (3)$$

In which H = (KH, RH, EH) and G = (KG, RG, EG), respectively be two keyphrase graphs that represent the query and document in question. Π = (*f, g*) is a projection from H to G defined in [6] and consists of two mappings f: RH → RG, g: KH → KG. $\alpha: K \times K \rightarrow [0,1]$ and $\beta: R \times R \rightarrow [0,1]$ measure semantic similarity between two keyphrases and two relations defined in the ontology model, *W(k)* is the weight of keyphrase in query, defined in section 4.2. The valuation function $v_w(\Pi)$ can also be used on a partial projection from sub-keyphrase graph H' of H to G. The weighting is pass on intact from H to H', therefore with the number of keyphrase reduce, the sum of all *W(k)* in H' is always smaller than 1, the evaluation of partial projection will never amount to 1. This is our intention so that the H' will be valuated higher if its contain keyphrase with high weighting. To determine relevance between a document and user query is to calculating the relevance betwwen two keyphrase G and H represent them with $rel(H,G)=Max(v_w(\Pi) \mid \Pi$is a partial projections from H to G).

V. Experiment and Evaluation

We evaluate the effectiveness of our techniques using the two classical measurements for a retrieval system, the recall and precision. First, we built ontology for the Computer Science domain and collected more than 10,000 documents, mostly papers, e-book and some theses. Those documents are used to create several experimental document retrieval systems as per SDB model, the size of which varying from 1000, 2000, 5000 to 10,000 documents. We test each system with 100 queries and calculate then average recall and precision over those 100 queries. The result is compared with the performance of the old techniques and record in the below charts:

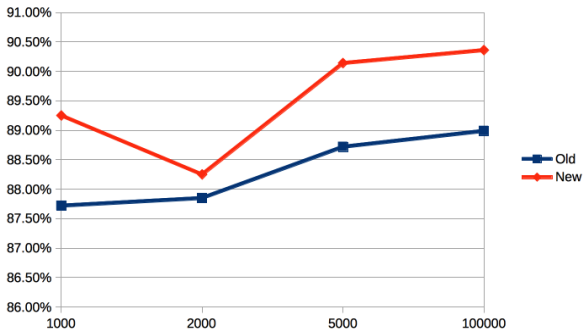

Figure 1. Recall comparison between techniques.

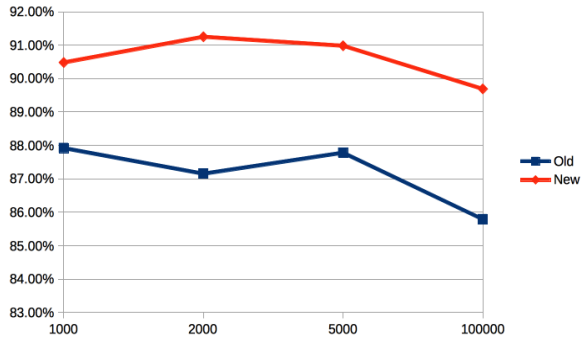

Figure 2. Precision comparison between techniques.

It can be seen that there was a keen increasing in precision while using the technique presented in this paper over our previous work. In average of all four testing collections, we were able to boost the precision from 87.16% to 90.6%. The recall measurement while also increase was not as sharp. The average recall among the four collections was raised from 88.32% to over 89.5%.

VI. Conclusion

In this paper, we have shown some improvements for the document retrieval system design that supports semantic representation and processing in search. A

system of rules has been added to the ontology model to standardize measuring semantic relatedness between keyphrases. Moreover, the techniques for semantic search are also improved effectively thanks to adding weighting in query's keyphrase graph to supporting query processing and upgrading the formula of measuring similarity between keyphrase graphs more precisely. These improvements result in the better semantic representation of the ontology and the higher average precision and recall of the system which has been implemented and tested at University of Information Technology, Ho Chi Minh city, Vietnam. Therefore, researching and developing our solution for document retrieval systems have been being the basic for many specific resource management systems in various different information domains.

ACKNOWLEDGMENT

This research is the output of the project "Researching and developing the semantic - based solution for Learning Resources Management" under grant number D2014-02 which belongs to University of Information Technology -Vietnam National University HoChiMinh City.

REFERENCES

[1] D. Sánchez and M. Batet. "A semantic similarity method based on information content exploiting multiple ontologies," *Expert Systems with Applications*, vol. 39, no. 9, pp. 1393-1399, 2013.

[2] H. Eriksso, "The semantic-document approach to combining documents and ontologies," *International Journal of Human-Computer Studies*, vol. 65, no. 7, pp. 624-639, 2007.

[3] M. Fernández, I. Cantador, V. López, D. Vallet, P. Castells, and E. Motta, "Semantically enhanced information retrieval: An ontology-based approach," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, no. 4, pp. 434-452, 2011.

[4] D. Sánchez, A. Solé-Ribalta, M. Batet, "Enabling semantic similarity estimation across multiple ontologies: An evaluation in the biomedical domain," *Journal of Biomedical Informatics*, vol. 45, no. 1, pp. 141–155, 2012.

[5] S. Kara, Ö. Alan, O. Sabuncu, S. Akpınar, and N. K. Cicekli, "An ontology-based retrieval system using semantic indexing," *Information Systems, Journal Information Systems*, vol. 37, no. 4, pp. 294-305, 2012.

[6] V. N. Do, T. T. T. Huynh, T. P. Nguyen, "Semantic representation and search techniques for document retrieval system," *Intelligent Information and Database Systems*, vol. 7802, pp. 476-486, 2013.

[7] M. Chein and M. L. Mugnier, *Graph-based Knowledge Representation*, Springer, 2009.

[8] M. S. Zhong, "Indexing conceptual graph for abstracts of books. *Fuzzy Systems and Knowledge Discovery (FSKD)*," in *Proc. 2011 Eighth International Conference on Volume 3*, 2011, pp. 1816-1820.

[9] S. S. Kamaruddin, "Dissimilarity algorithm on conceptual graphs to mine text outliers," in *Proc. 2nd Conference on DMO '09*, 2009, pp. 46-52.

ThanhThuong T. Huynh received her B.Sc degree - Honor Program (2007) follow by M.Sc degree (2012), both in Mathematics and Computer Science, from University of Science - Vietnam National University Ho Chi Minh City. She also has BA on Bussiness Administration (2010) from University of Economics Ho Chi Minh City and currently doing a PhD course in Computer Science at UIT.

Upon graduation, she worked as a Senior Software Developer and Project Manager for several corporations. She is currently a lecturer at the Faculty of Computer Science, University of Information Technology (UIT) - Vietnam National University Ho Chi Minh City, teaching courses related to Artificial Intelligence, Knowledge Engineeringe and Machine Learning. Her research interests include Knowledge Representation and Reasoning in Artificial Intelligence, Intelligent Systems, Knowledge Base Systems, and most recently, Semantic Search Engines and Information Retrieval Systems. She has participated as a key member in various research projects and publications related with Semantic representation and search techniques for document retrieval and management systems. She has leaded several research projects such as the very project that made this article possible.

TruongAn PhamNguyen graduated with honor in 2011 then go on to complete a Msc. course both in Computer science at University of Information technology, Ho Chi Minh city, Vietnam. Upon his graduation, he is employed as a lecturer at Computer science faculty, University of Information technology.

Hung K. Chau is a lecturer in Computer Science Department at the University of Information Technology. He holds a BS degree and a MS degree from the University of Information Technology, HoChiMinh, Vietnam. His research interests are Information Retrieval, Data mining, and Knowledge representation and Reasoning

Nhon V. Do is currently a senior lecturer in the faculty of Computer Science at the University of Information Technology, Ho Chi Minh City, Vietnam. He got his MSc and Ph.D. in 1996 and 2002 respectively, from The University of Natural Sciences – National University of Ho Chi Minh City. His research interests include Artificial Intelligence, computer science, and their practical applications, especially intelligent systems and knowledge base systems.