A Subject-Oriented Ontology Development for Information Retrieval Application

Ta Cong Duy Chien [(1)], **Phan Thi Tuoi** [(2)], **Nguyen Chanh Thanh** [(3)]

[(1, 2)] HoChiMinh City University of Technology, 268 Ly Thuong Kiet HCM City, Vietnam

[(3)] HUTECH University of Technology, 475A Dien Bien Phu Street, HCM City, Vietnam

Email: chientdc@cse.hcmut.edu.vn, tuoi@cse.hcmut.edu.vn, nc.thanh@hutech.edu.vn

Abstract - Recently, ontology-based application is an important approach of some researches in several fields of Computer Science. The development of Subject-Oriented Ontology (SOO) may be hard for research groups. However, since SOO can apply to various areas especially for subject-driven researches, research groups are continuing to search for solutions for building it. The paper proposes an approach to develop an SOO based on corpus of scientific papers by building subject trees and semantic relationships among them. Our experiments were tested on the ACM corpus and we have acchieved good results in the first phase.

Keywords - Subject-oriented Ontology, Subject-driven Ontology, Subject Tree.

I. INTRODUCTION

There are many smart systems in the world, which can provide advanced features to meet user's requirements in specific cases such as "*finding eBooks about Semantic Web or extracting all data relating Compiler*", etc. It may be a next smatter generation of existing Information Retrieval and Question Answering systems. There may be many different approaches to building these systems, and one of them is to build an SOO. An SOO can provide the information, which is related to subjects expected by users. In order to answer the complex requests from users, the researchers had taken their good times to build SOO with high quality.

By analyzing data on social networks, we have found that users usually have many kinds of different needs, such as using search engines, question-answering system, extracting data relevant to business, learning, etc. Therefore, it is important to have a smart system as above-mentioned, which can satisfy users, especially those who use search engines and/or digital library systems to search for necessary information regarding their predefined subjects. Hence, the idea of building an ontology with "*subject-driven*" (sometimes it call subject-oriented ontology, SOO) is still on top focus of researches up to date.

Initially, to build that SOO, our approach was focused on categorizing members or instances of ontology into some groups based on Support Vector Machine (SVM) technique [1]. However, it was not our best choice due to complicated calculations and huge efforts for preparing training data set. Therefore, to solve these problems, we applied the tools of Natural Language Processing to build it.

Another primary task in our research is to explore how to enrich a SOO that users can retrieve many kinds of its information such as group of relevant trees with the same subjects or the same group of keywords, hierarchical trees regarding a given subject, etc. Based on this task, the SOO can apply to develop some applications as follows:

- A subject-oriented information searching system
- A subject-oriented information extraction system
- Information/Document categorization

For example, when users entered a query, it will be parted into some of keywords that can be linked with one or many keywords in space L_2 of SOO and subject trees in space L_3. Those trees can provide more relevant subjects, which may be selected as primary subjects for this query. They can also list out more relevant keywords which may be used in expanding search process (in case of information searching system) or extracting data process (in case of information extraction system).

The paper would introduce how that selected approach can be followed up to develop an SOO with its four-layer structure with different roles and many nouns, compound nouns, as described in Fig 1 of section 3.

In this paper, the overview of our research is described in section 2. Section 3 includes introductions about characteristic of SOO and how it can be enriched based on relevant heuristics. Other sections are dedicated for presenting our experiments that many practice steps were performed on the corpus of scientific papers referred in ACM Digital Library website [2], experimental evaluation and discussion of conclusion and future work.

II. RELATED WORKS

In our initial review, it seems that there is no published result on Internet about SOO development or ontology based on integrations of subject trees. However, there is some "*similar*" ontology such as topic-oriented ontology or topic ontology.

First of them is the research of Ana's group [3]. They built the web topic ontologies by classifying Web pages content into groups with different topics. After that, they re-organized them into a hierarchical scheme and build cross-references in kinds of "*is-a*", "*symbolic*" and "*related*" between different topics in a non-hierarchical scheme. There are two differences between this approach and our approach. The first, subjects in the above research are predefined in open directory [4] but subjects in our approach are dynamically recognized in titles of papers; the second, their cross-references in three kinds like above are not linked to WordNet and, hence, they cannot extend their subjects but our approach does not face that issue.

Another study is Bhavani's research [5] in which they had clearly different layers of learning resources/occurrences, topics and their associations. All of those layers including elements are built into network of topics and internal links between many topics. Similarly, to [3], they did not have any extension because they did not link to WordNet [6].

Next is Xujuan's research [7] that they defined a domain ontology based on user profile and its semantic relationships such as "synonymy" and "*hyponymy*". The key points are that the subjects are comprised from terms extracted in documents, including primitive classes and compound classes. Our approach is similar to this approach; however the difference is that subjects of ontology in our approach are in form of noun phrases and extracted from titles of documents.

The research of Roberto team [8] had other view of domain ontology via decentralizing ontologies of different domains. The strong point of [4] was that its topics are flexibly gotten from the names of companies, user profiles and interests, content of the projects, etc. There were also many kinds of its relationship, such as Synonym, Hyponym, Hypernym, etc. However, they did not link to WordNet [6] and this may be a weak point of the research.

The researches of Blaž's team [9] and [10] provided some important ideas such as clustering documents to find nodes in the domain ontology, then extracting terms and keywords from clustered documents. After that they created tree-based concept hierarchy which can be presented as a form of the ontology.

The last research is performed by Tuoi et al [11]. In this research, the Vietnam Knowledge Base (VKB) was developed based on the structure including elements such as Class, Object and relations Roo, Roos, Rcc, Rcch and Roc among instances of VKB. The advantage of VKB is that its structure is good because it can contain all information of scientific papers in ACM, IEEE or other data sources. However, those subjects do not include information of paper subjects and they are linked to WordNet. Therefore, VKB structure must be upgraded to become a SOO.

III. THE SOO DEVELOPMENT

A. A proposed structure of SOO

By analyzing the structure of a scientific paper in ACM Digital Library [2] with special defines in some parts of that structure, we can extract the necessary information to build a SOO. In our proposal, the structure of SOO is defined as SOO = [*D*, *I*, *T*, *N*] in which:

D: the set of documents (d_1, d_2 … d_n) from a given corpus. Actually, those documents are papers in plain text format. They can be retrieved from a website of ACM Digital Library, Springer or IEEE.

I: the set of items (in tuple <*value*, *frequency*>) extracted from documents in *D* as $I = [K=\cup K_i, S=\cup S_i, C=\cup C_i]$ (i=1..n) with K_i , S_i , C_i are the set of keywords, subjects and classes extracted from document d_i in *D*.

T: the set of subject trees $t^i_j = [s, K_i, C_i, L_i]$. As illutration in Fig 1.a, t^i_j includes a subject s (in S_i) as the root node, keywords k^i_j (in K_i) as leaf nodes, and

mapping to the lowest level of classification tree (built from C_i). L_i is the set of 2-ways links $l^i_j = <s, k^i_j, p_{sk}, p_{ks}>$ from s to k^i_j with probability p_{sk} representing k^i_j's happening with given s and probability p_{ks} representing s's happening with given k^i_j. Besides, *T* also includes links among s_i and c_i with probability p_{sci} as follow: $l_i=<s_i,c_i,p_{sci}>$

N: the semantic network among trees in *T*, which includes links represented in tuple <*relation*, *probability*> with relation belongs to WordNet's semantic relations. The relations are included synonym, hyponym, hypernym and the WordNet is version 2.0 in case.

The above statements describe SOO's components and structure. Besides, the structure of SOO is also represented in four layers as illustration in Fig 1.b, in which L_1, L_2, L_3, L_4 are *D*, *I*, *T*, *N* respectively and their combination will provide a complicated structure of SOO. However, it can consist of all necessary information of papers in [2].

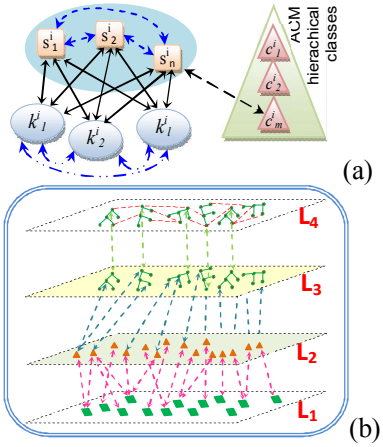

Figure 1. The subject tree's structure (a) and structure of SOO (b) built from subject trees

As mentioned above, L_1 is set of text documents of ACM Digital Library; L_2 is set of terms extracted from L_1; L_3 is set of topic trees; L_4 is set of semantic relations among trees in L_3.

Next step, we will propose 3 steps in order to build and enrich SOO from a given data source (Fig.2).

B. An approach to build and enrich a SOO

To build consecutively all of four layers of an SOO, there are three steps A, B and C as the below illustration. Each step will have each own target with different inputs.

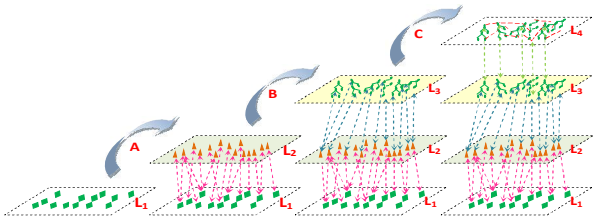

Figure 2. The development and enrichment approach to build SOO

We will describe those steps in details as below:

***Step A*: Extracting keywords, classes, subjects from a document in L_1**

The heuristic is based on the mapping: f_A: $L_1 \rightarrow L_2$ that $f_A(d_i) = <K_i, S_i, C_i>$

Here d_i is a document in *D* (as known as L_1), K_i is the set of keywords in d_i, S_i is the set of extracted key phrases in the title of d_i, C_i is the set of classes belonging to ACM hierarchical classes.

Procedure of Step A:

```
In: set of document D (L1)

Out: set of items I of keywords, subjects and hierarchical classes

Process:

K ← empty; S ← empty; C ← empty;

For each document di in L1

  K ← K ∪ (Ki = extract_keyword(di));

  C ← C ∪ (Ci = extract_class(di));

  S ← S ∪ (Si = extract_subject(di);

End for;

Update occurrence probability for each item in K, S and C;

Return I = [K, S, C] = [∪Ki, ∪Si, ∪Ci];

End procedure.
```

Extract_keyword is the simple function to extract text in attribute "*content*" of Meta data tag "*citation_keywords*" (defined in web content by [2])

Extract_class is also the simple function to extract text in "*primary classification*" and "*additional classification*" section (defined by [2])

Extract_subject is the most complex function which follows below heuristics:

Procedure of extract_keyword:

```
In: title of a paper t

Out: list of subject KP

Process:

Call GATE tool [12] to extract list of key phrase KP = {kp_i} from paper t;

For each kp_i in KP: if kp_i's tree is a subtree of kp_j's tree, remove kp_i out of the list KP;

End for;

Return KP;

End procedure.
```

***Step B*: Building subject trees in L_3**

The heuristic is based on the mapping: f_B: $L_2 \rightarrow L_3$ that $f_B(<K_i, S_i, C_i>) = \{ t^i_j \}$ and t^i_j is the subject tree with the structure illustrated in Fig.1(a).

Procedure of Step B:

```
In: set I = [K = ∪K_i, S = ∪S_i, C = ∪C_i] (i=1..n)

Out: set of subject tree T

Process:

T ← empty;

For each tuple <K_i, S_i, C_i> (i=1..n)

  For each j (j =1..|S_i|)

    Create t^i_j =[s^i_j, K_i, C_i, L_i] (subject tree) following structure in Fig 1.a;

    If t^i_j is duplicated with an existing tree t in T

      Add K_i to keyword set of t;

      Add C_i to class set of t;

      Delete t^i_j;

    Else

      T ← T ∪ {t^i_j};

    End if;

  End for;

End for;

For each tree t = [s, K_i, C_i, L_i] in T:

  With each L_i's link, update p_sk & p_ks: p_sk = 1/(|S|*|K_i|), p_ks = 1/(|K|*|∪S_i|(*));

End for;

Return T;

End procedure.
```

(*): the number of groups with trees which have links to given keyword.

***Step C*: Updating semantic relationships among subject trees and building semantic network in L_4 to create an SOO**

The heuristic is based on the mapping: f_C: $L_3 \rightarrow L_4$ that $f_C(T_i=\{ t^i_j \}) = <T_i, R_i>$

Here $<T_i, R_i>$ is the trees which have been updated for the relationship group R_i that is a subset of WordNet relationship set.

The combination of the layer L_1, L_2, L_3 and L_4 will become the ontology with structure as Fig 1.b.

Procedure of Step C:

```
In: set of subject tree T, set of relationship R_W from WordNet

Out: semantic network N

Process:

N ← empty;

For each subject tree t^i_j in T

  If its keyword is also keyword(s) of other trees, add to N an "associate"
```

relationship with probability of co-occurrence between t^i_j and those trees;

If its keyword has a relationship (**) with keyword(s) of other trees, add to N an "associate" relationship with probability of co-occurrence between t^i_j and those trees;

If its subject has a direct (or indirect) relationship (***) with the subject of other trees, add to N same kind relationship with probability of co-occurrence between t^i_j and those trees;

End for;

End procedure.

(**): it can be "*similar*", "*hyponym*" … as defined by WordNet

(***): there is a linked path through several internal nodes in WordNet to connect from a subject to another subject.

After this step, the last layer (L_4) with the semantic network *N* is built based on links among subject trees in L_3 and WordNet. As such, the SOO is also developed based on component *D*, *I*, *T* and *N* as its structure presented in above.

IV. EXPERIMENT AND EVALUATION

At the beginning, to prepare testing data for our experiment, an English corpus was built based on 65,112 documents. Each of documents is a plain text file and includes the brief descriptions of the full paper in webpage format from ACM's resource.

After that, the experiments of steps A, B, C were conducted in below steps.

A. Step 1: Verifying data set to find valid documents

Initially, our analysis was performed on the corpus with following results:

- Group A with 21,433 documents (32.92%) having no keywords.
- Group B with 34,679 documents (53.26%) having one or many keywords.

In detail, documents in Group A can be divided into three kinds:

- The first with 360 documents describing the introduction or table of content of proceedings and journal.
- The second with 5,475 documents containing incompletely content caused by errors during downloading progress.
- The last with the remaining documents which relevant source (web pages) do not include any keyword.

After that, those documents in Group A were investigated to detect "Primary Classification" (this is the section listed out hierarchical classes defined by ACM) with 19,419 cases having no information of Primary Classification [13].

The same investigation was performed on Group B and it shown that Group B's documents were sufficiently good for our working in the next steps. Therefore, Group B is selected as the official resource data for our experiment in the below steps.

B. Step 2: Extracting keywords, classification classes and titles of documents

Based on 34,679 documents of Group B, the keyword extraction step was done with the result of 98,502 keywords (in brief: kws) extracted from all documents as in Table 1.

However, the quality of this result wasn't so good because some errors happened when the maximum number of keywords that users can enter is limited. It has impacted the total number and the correctness of retrieved keywords. Our next verification found that only 37,933 keywords were correct enough for using in next steps. As such, the precision of our work is only 38.51%. In addition, our work faced the issue of wrong keyword detection in several cases such as keywords including special characters or symbols, keywords with non-alphabet characters. The statistic in Table 1 also shown that most of documents had 2 to 4 keywords per document.

Table 1. The summary of Group B's documents grouped by number of keywords

#kws per document	***2 kws***	***3 kws***	***4 kws***	***5 kws***	***Error cases***
Total documents	9,686	7,396	4,428	2,301	1,998
Total keywords of documents	19,372	22,188	17,712	11,505	27,725

After that, the classification class extraction was done with good result of 100% precision and total 104,037 classification classes. The main reason was that every document had primary classes as required by ACM. Moreover, all author's entered classes in each paper were selected from the list defined by ACM without any exception for manual update. That explained why our step achieved the highest number. However, our review detected that those classes were actually duplicated and there were 425 unique ACM classes that followed the structure of ACM Computing Classification System [13].

The last work was to retrieve all titles of the documents of Group B. Luckily; each paper had a unique title on the top of document. Therefore, Group B provided the list of 34,679 titles without any issue during our processing.

C. *Step 3: Extracting key phrases from documents' titles, building subject trees.*

Next experiment was the subject extraction step to recognize key phrases (in brief: kps) in document titles. Many researchers working on kps usually considered it grammatically in kind of noun phrases (NP). Our work was similar to those for key phrase extraction by focusing on NPs in all Group B's document titles. In this step, GATE toolkit was used to detect and retrieve NPs from documents. Below were some key results of this step:

There were 49,219 key (noun) phrases extracted from 34,679 documents of Group B. However, only 35,837 unique valid key phrases were selected relating to 33,404 titles of the total 34,679 titles (~96.32%). Therefore, the precision achieved 72.81%.

The invalid key phrases have occurred because of some reasons such as unreasonable contents or wrong extraction on titles that happened in 1,275 documents (~3.68%).

Table 2. The detail result categorized by group of key phrases

#kps per title	***1 kps***	***2 kps***	***3 kps***	***Error cases***
Total titles	19,453	12,303	1,505	143
Total key phrases of titles	19,453	24,604	4,515	645

As shown in Table 2, many documents had only one or two key phrases and the 645 uncorrected cases (~1.31%) which were related to 143 titles of the 1,275 documents were removed out of this analysis due to some reason such as their contents not making sense, noise or wrong extraction on titles. Therefore, only 35,837 key phrases (~98.68%) were kept in the list.

In summary, this step produced 35,837 unique valid key phrases and revised the Group B by keeping just 33,404 documents. All of them were selected for inputs for subject tree development step with 35,837 subject trees were built based on 35,837 "*good*" key phrases, 37,933 "*good*" keywords and 425 ACM classes without any error (because of following defined simple structure in Fig 1.(a).

D. *Step 4: Identifying mappings among trees & WordNet, building ontology*

The most difficult task was to recognize the best relationships between subject trees and WordNet, and then to assign them to pair of subjects. The result was achieved as follows:

There were 3,336 subjects (in the root of subject trees) to have connections directly to word/sense list in WordNet. Therefore 3,336 direct links to WordNet were determined and built in this step. However, only 1,414 of them were the valid links. A valid link is an existing link between subject trees and WordNet. This shown a precision level of ~42.39%. All of these links were in kind of "associate" relation defined by WordNet.

The remaining subjects (32,501 of 35,837) could not be directly mapped to WordNet, however they could have internal connections to 1,414 trees as above in relation kinds of "*associate*", "*similar*", etc., which connections could be (external) indirect links to WordNet. This experiment found 42,293 indirect links to WordNet from these subjects, but only 32,656 of those links were correct and unique for our selection because the others were invalid or duplicated. Therefore, the precision here was 77.21%. In addition, in order to check a valid link, we use sequence language for querying on WordNet's database.

With the results retrieved from above works, they were good enough for our developing an SOO (as known as subject-driven ontology). In order to build this ontology, we used some tools such as:

- Protégé to define the structure and mapping of the ontology

- GATE to perform testing on built results.

Finally, our ontology was built based on Group B's documents and it consisted of:

- Total 35,837 subject trees including 37,933 keywords with 425 classes, 1,414 direct links to WordNet.
- 32,656 indirect links among subjects of the ontology.

After this step, our target was achieved with the SOO and its components.

E. Step 5: Evaluating SOO

In the initial scope, there is no plan to perform experiment for utilizing ontology to support other application; therefore, there was not an evaluation for that task here. Besides, Word Sense Disambiguation (WSD) in SOO was not taken to analyze and implement. They should be the important key goals for our further research.

Back to the results that we have gotten from above steps, there were several difficulties happened in our experiments:

- The first thing is related to the quality of data source, which are used to enrich the SOO. Although all documents in our dataset (Group B) were made based on corresponding papers on ACM Digital Library web site, some of them were actually not in good format. This one impacted the result of next practice steps.
- The second thing is that out-standing issues during working on syntactic detection. We had some of troubles which are related how to make sense of keywords and key phrases
- The third thing is related to multi-lingual. Until now, our top priority is just English document. However, there are many other languages that can be focused on and processed.

V. CONCLUSION AND FUTURE WORK

Our paper has presented an approach how to develop SOO based on subject trees. They are also linked (directly or indirectly) together to WordNet. In general, SOO has wide range of applications in some re-searches regarding subject-oriented, such as searching, extracting or categorizing information by subject, etc. The experiment shown that our SOO, which was built and enriched with data gotten from ACM data source. However, its quality and quantity are not so good because of existing issues in practice steps. If they can be completely solved, the quality of SOO would have better. They will be the next target for our research to "*optimize*" logically and practically this approach. Our next research will also focus on WSD in SOO by applying "*probability approach*" and build a framework for applications based on this SOO. All of these goals may be hard to achieve, but we believe that they will make significant contributions to NLP and Semantic Web community once they are realized.

VI. ACKNOWLEDGMENTS

Our thanks to all members in the BK-NLP group of HCM City University of Technology Vietnam for their enthusiastic collaboration, also thanks to HCMC National University in their support of our key project "*Research and develop an Q&A and Information Retrieval system with supporting Vietnamese language for Digital Library*" of HCMC National University.

REFERENCES

[1] "SVM - Support Vector Machines" [Online]. Available: www.support-vector-machines.org .

[2] "ACM Digital Library" [Online]. Available: dl.acm.org.

[3] A.G, Cechini et al, "Using Topic Ontologies and Semantic Similarity Data to Evaluate Topical Search," in Proceedings of 36th Latin American Informatics Conference (CLEI), Asuncion, Paraguay, 2010.

[4] "Open Directory Project" [Online]. Available: www.dmoz.org.

[5] Bhavani et al, "An ontology-driven topic mapping approach to multi-level management of e-learning resources," in Proceedings of 17th European Conference on Information Systems (ECIS), Verona, Italy, 2009.

[6]"WordNet" [Online]. Available: wordnet.princeton.edu.

[7] Xujuan et al, "Utilizing Search Intent in Topic Ontology-Based User Profile for Web Mining," in Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, Hong Kong, China, 2006.

[8] Roberto et al, "Semantic Navigation through Multiple Topic Ontologies," in Proceedings of Semantic Web Applications And Perspectives, the 2nd Italian Semantic Web Workshop, Trento, Italy, 2005.

[9] Blaž et al, "Proceedings of the 9th International multi-conference Information Society IS-2006," Ljubljana, Slovenia, 2006.

[10] Blaž et al, "Semi-automatic construction of topic ontology," in Proceedings of the 8th International multi-

conference Information Society IS-2005, Ljubljana, Slovenia, 2005.

[11] Tuoi, T.P et al, "Vietnamese Knowledge Base development and exploitation," International Journal of Business Intelligence and Data Mining, vol. 6, no. 1, 2011.

[12] "GATE, A General Architecture for Text Engineering" [Online]. Available: gate.ac.uk.

[13] "ACM Computing Classification System" [Online]. Available: www.acm.org/about/class/ccs98-html.

AUTHORS' BIOGRAPHIES

Ta Cong Duy Chien is Ph.D. Student at the Faculty of Computer Science and Engineering, HCMC University of Technology, Vietnam. His research interests include Natural Language Processing and its applications, Information Extraction.

Nguyen Chanh Thanh graduated from HCMC Pedagogics University as BS of Mathematics, 1994. After that he received his BE of Information Technology, ME of Computer Science and DSc of Computer Science 1998, 2003 and 2011 respectively from HCMC University of Technology, Vietnam. His research interests include Information Retrieval/ Extraction, Semantic Web, Natural Language Processing.

Phan Thi Tuoi is a Professor at the Faculty of Computer Science and Engineering, HCMC University of Technology, Vietnam. She obtained her Ph.D. in Computer Science from Charles University, Czech Republic, in 1985. Her research interests are compiler, information retrieval, natural language processing. She has been the Chief Investigator of national key projects and published many papers in international journals and conference proceedings in those areas