A Study of Thai Succession Law Ontology on Supreme Court Sentences Retrieval

Tanapon Tantisripreecha and Nuanwan Soonthornphisaj

Abstract—**This paper presents an improvement of our approach called SCRO_II algorithm. SCRO_I algorithm was initially developed in order to retrieve a set of Supreme Court sentences. The goal of SCRO_I is to provide different law issues among those retrieved documents. We create a new ontology using different semantics to study their performances based on diversity measurement. The contribution of this new ontology is compared to the traditional one. A new procedure is embedded in SCRO_II algorithm to identify a set of synonyms and relations. The experimental results show that using the new ontology as a knowledge representation can enhance the retrieval performance of the algorithm. We study the retrieval performance based on diversity value and found that SCRO_II outperforms SCRO_I algorithm.**

Index Terms—**Ontology, retrieval, Supreme Court sentences, Thai Succession Law.**

I. Introduction

Decisions made by judges are very important for users in legal fields because these legal documents provide some legal doctrines that can not be found in the code laws. Studying Thai law using only code laws is not possible to fulfill the knowledge of those users. The Supreme Court of Thailand had provided a search facility through the web site (Deka System) (http://deka2007.supremecourt.or.th/deka/web/search.jsp). The corpus of the current system consists of 100,010 legal documents. These documents are not predefined into categories of law domain, therefore the retrieval performance is not satisfied in terms of semantic diversity of law content presented in the documents. The reason that legal users need to obtain the variety of law issues from the retrieved document is that they need the complete understanding of each law area in all aspects. To tackle this problem, we propose to use an ontology as a knowledge representation and develop an algorithm called SCRO_I to generate a set of keywords that guarantee the document diversity [1]. However, the performance measured in term of document diversity obtained from SCRO_I algorithm is only 54.6%. The objective of this paper is to improve the performance of SCRO_I algorithm. We propose a new ontology model that has an ability to enhance the retrieval performance. Another benefit of our proposed algorithm is that users don't need to input several keywords for searching since the algorithm will automatically create a set of keywords for the retrieval process.

This paper is organized as follows. Section 2 describes an overview of ontology and our proposed ontology. Section 3 addresses the SCRO_I and SCRO_II algorithm. Section 4 describes the evaluation details of ontology-based retrieval system. Finally we conclude our work with some discussion and directions of future work in Section 5.

Manuscript received December 30, 2009.

Nuanwan Soonthornphisaj is with the Department of Computer Science, Faculty of Science, Kasetsart University, Bangkok, Thailand. E-mail: fscinws@ku.ac.th, Phone no. +6689-2007553. (Corresponding Author)

Tanapon Tantisripreecha is with the Department of Computer Science, Faculty of Science, Kasetsart University, Bangkok, Thailand. (e-mail: g5164138@ku.ac.th).

II. Ontology

A. Related Works on Ontology

Many domain specific ontologies were constructed in various areas such as medical domain [9], e-learning domain [10], legal domain [1]. Ontologies were applied to enhance existing applications such as knowledge-based system [2], computational linguistics [3], etc. Benjamin, *et al*. [6] applied ontology to improve IT support for judges in Spain. They used questionnaires as a tool to collect requirement from users in order to implement the ontology. E-commerce law ontology was constructed using text mining technology by Kayed [7]. A set of conceptual nodes were extracted from e-commerce law cases. They proposed a new algorithm to reduce a number of links connected between nodes in order to decrease the ontology complexity.

Legal Information retrieval research was done for French law [4]. Saravanan, *et. al.* did research about query expansion for Indian law in three area which are rent control, income tax and sale tax. They compared their proposed algorithm to the keyword search and found that their algorithm outperformed the traditional retrieval algorithm [5]. The first Korean law ontology was generated using natural language processing technique cooperated with Korean WordNet. The dataset was collected from law text books and research papers about law. After that, their algorithm found a set of word co-occurrences and hand on ability to construct a graph that connects these words together [8].

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

B. *Thai Succession Law Ontology*

We set up a structure of the Thai succession law ontology using a set of relations and a set of synonyms. Our ontology has two levels, a core level and an extended level ontology.

The core level ontology represents the knowledge of Thai succession law. Fig. 1 shows an example of ontology. In case that user enters keyword, 'r_1', SCRO_II algorithm traverses through the core level ontology and found the relation 'r_1' exists. Next, the algorithm explores all links attached to node between 'r_1', (r_1*, r_2, r_3, r_4). The star sign (*) found in r_1 means that there is an extended knowledge related to r_1. Therefore the algorithm explores the extended level of r_1 and found word1, word2 and word3. Finally a set of queries (Q) are obtained as {A, r_1, B}, {A, word1, B}, {A, word2, B}, {A, word3, B}, {A, r_2, B}, {A, r_3, B} and {A, r_4, C}, respectively.

In case that the keyword obtained from user is found in the node (eg. 'A') of ontology, SCRO_II traverses through the ontology and generates the set of queries as follows: {A, r_1, B}, {A, r_2, B}, {A, r_3, B}, {A, r_4, C}, {A, r_1, S_1}, {A, r_2, S_1}, {A, r_3, S_1}, {A, r_1, S_2}, {A, r_2, S_2}, {A, r_3, S_2}, {A, r_1, S_3}, {A, r_2, S_3}, {A, r_3, S_3}, respectively.

We develop an application for a law expert to create Thai succession law ontology. First, the conceptual nodes were created when user enter keywords into the system. Then the links were connected when user enter words that represent the relationship between nodes. The synonym and reason relation are automatically created using ThaiLegalWordnet.

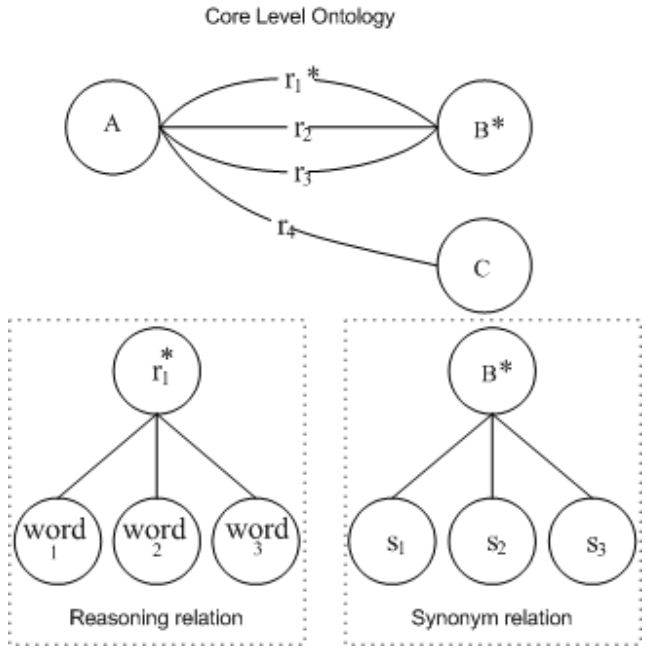

Fig 1. The proposed framework of ontology is divided into core level and extended level.

III. ALGORITHMS

A. *SCRO_II algorithm*

The framework of SCRO_II algorithm is shown in Fig.2. Given a keyword obtained from user, SCRO_II searches through the core ontology for the keyword. If the word is found in the node of the ontology, the algorithm will traverse through the link to the next node. The words occurred in the nodes and relations are generated to form a set of query.

In case that user enters keyword, 'A', SCRO_I algorithm traverses through the ontology and found the node 'A'. Next, the algorithm explores all links attached with node 'A', (r_1, r_2, r_3, r_4, word1, word2,word3). Then a set of queries (Q) are obtained as {A, r_1, B}, {A, word1, B}, {A, word2, B}, {A, word3, B}, {A, r_2, B}, {A, r_3, B} and {A, r_4, C}, respectively.

SCRO_II also explores the extended level of ontology under two different conditions. In case that the keyword is found in the node, the extended level ontology will be explored only when the star sign is found in the node.

Table 1. SCRO_II algorithm

Algorithm : SCRO_II

Input: Ontology, keyword
Begin
Word = {}
Query = {}
ConceptType = {NODE, RELATION}
SearchConcept (Ontology, keyword)
Word = OntologyTraversal (keyword)
For Each w_i in Word Do
 If ConceptType(keyword) = NODE and ConceptType(w_i) = NODE
  If FoundExtendedOntology(w_i)
   ExtendedWord = ExtendedOntologyTraversal()
  If ConcepType(w_i) = RELATION and ConceptType(w_i) = RELATION
  If FoundExtendedOntology(w_i)
   ExtendedWord = ExtendedOntologyTraversal()
End For

Query = QueryGenerate(Word,ExtendWord)
For each q_i in Query
 D_i = DocumentRetrieval (q_i)
 Ranking(D_i)
End For
CombineResults = checkDuplicate(D)
End

Output: CombineResults

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

On the other hand, if the keyword is the relation found in the ontology, the extended ontology will be explored when the star sign is found on that relation. Then the set of queries are created and used in the retrieval process. Considering a set of retrieved documents, the SCRO_II algorithm ranks these documents using TFIDF formula (see equation 1)

$$w_{in} = tf_{in} \times idf_i \quad (1)$$

where w_{in} = weight value of $word_i$ in the retrieved $document_n$
tf_{in} = the number of occurrences for $word_i$ in $document_n$

$$idf = ln\left(\frac{N}{n_i}\right) \quad (2)$$

where N = the number of retrieved documents
n_i = the number of retrieved documents contain $word_i$

SCRO_II algorithm selects the first ranked document as an output for each query and checks for duplicated documents. Since, each query (q_i) is processed independently, therefore the duplicate checking process is done to finalize the output. Because the objective of SCRO_II algorithm is to ensure the diversity of law issues from the final set of retrieved documents.

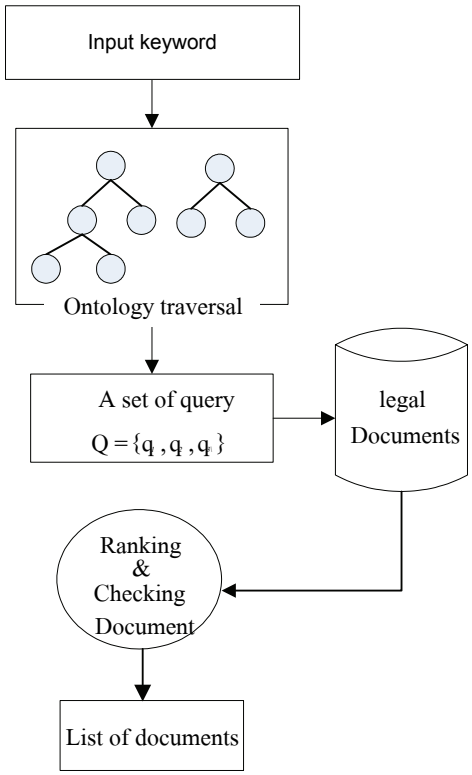

Fig 2. The framework of SCRO_II algorithm.

B. SCRO_I algorithm

The ontology used in SCRO_I algorithm has only one level. Given a keyword, the algorithm explores through the ontology and generates a set of query. The set of query obtained from SCRO_I is different from SCRO_II that yields different performance measure in term of diversity.

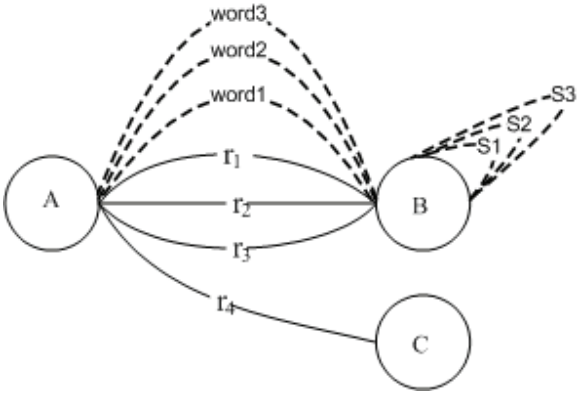

Fig 3. The structure of ontology in SCRO_I

In case that the keyword obtained from user is found in the node 'A', SCRO_II traverses through the ontology and generates the set of queries as follows: {A, r_1, B}, {A, r_2, B}, {A, r_3, B}, {A, r_4, C}, {A, $word_1$, B}, {A, $word_2$, B}, {A, $word_3$, B}, respectively.

If user searches input r1, the set of queries obtained from SCRO_I is {A, r_1, B}, {A, r_2, B}, {A, r_3, B}, {A, $word_1$, B}, {A, $word_2$, B}, {A, $word_3$, B}, respectively.

Table 2. SCRO_I algorithm

Algorithm : SCRO_I

Input: Ontology, keyword
Begin
Word = {}
Query = {}
Word = OntologyTraversal (keyword)
Query = QueryGenerate(Word, ∅)
For each q_i in Query
D_i = DocumentRetrieval (q_i)
Ranking(D_i)
End For
CombineResults = checkDuplicate(D)
End

Output: CombineResults

C. Baseline Algorithm and Deka system

We compare the performance of SCRO_II algorithm with the baseline algorithm and Deka system. The baseline system uses term frequency and inverse document frequency to determine the retrieved document. Whereas the Deka system (Fig. 4) simply uses SQL statements to retrieve legal documents containing a keyword. The current system

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

implemented by the Thai Supreme Court can be accessed via the website www.deka2007.supremecourt.or.th.

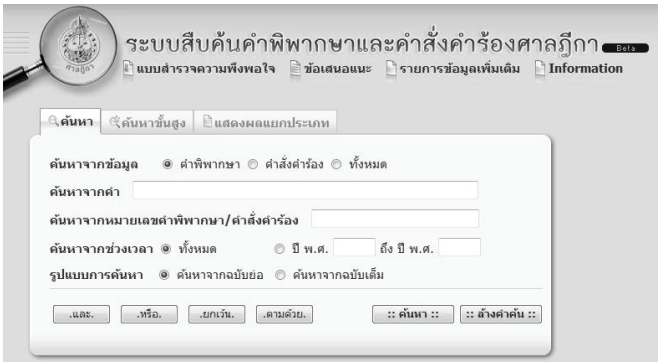

Fig. 4. The Supreme Court Sentences Retrieval System (Deka)

IV. Experiments

A. Data set

We collect all judge sentences related to Thai succession law. There are 248 documents in the corpus. The word segmentation is performed since Thai language has no explicit word boundary.

B. Performance measurement

Given a keyword, a set of retrieved documents are measured based on the degree of diversity coverage (DC). The maximum degree of diversity is 1. It means that the algorithm can retrieve all different law issues related to the keyword.

$$DC = \frac{\text{No. of law issues from the set of retrieved documents}}{\text{Total no. of law issues found in ontology}} \quad (3)$$

To test the performance, we compare the retrieval performance of SCRO_II to SCRO_I, baseline algorithm and Deka system, respectively.

In order to evaluate the retrieval performance on Thai succession law, we set up the experiment using 12 keywords which are are **มรดก** (estate), **ผู้จัดการมรดก** (administrator of an estate), **ทายาท** (heir), **ผู้ตาย** (deceased), **พยาน** (witness), **พินัยกรรม** (wills), **เจ้าหนี้** (creditor), **กำจัดมิให้รับมรดก** (exclusion for the succession) **สละมรดก** (renunciation of an estate) **พินัยกรรมไม่สมบูรณ์** (incomplete will) **ชำระหนี้** (performing loan) **ตั้งผู้จัดการมรดก** (appointment of the administrator of an estate), respectively.

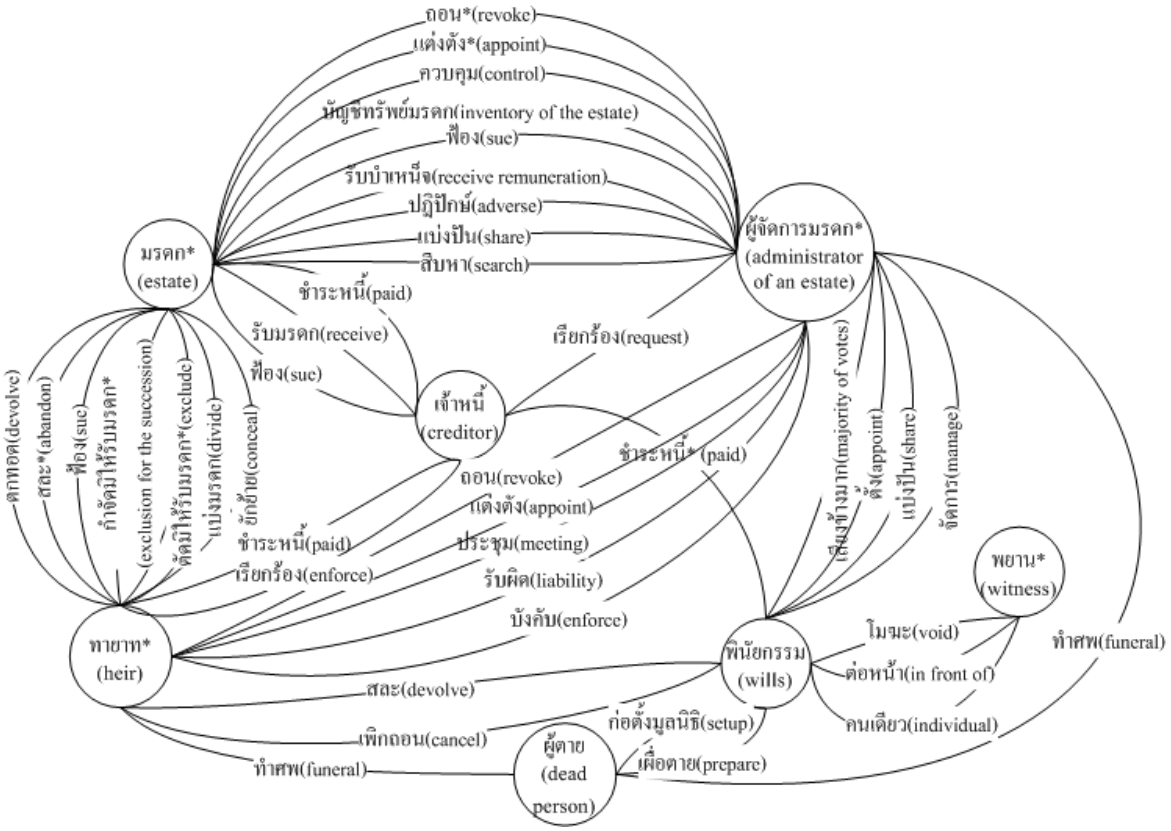

Fig. 3 Thai Succession Law Ontology

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 3. The comparison of performance in term of the number of law issues found by algorithms

	The number of	The Number of law issues found by algorithms			
keyword	predefined Issues	SCRO II	SCRO I	Baseline	Deka
มรดก (estate)	19	14	8	8	10
ผู้จัดการมรดก (administrator of an estate)	19	10	10	6	6
ทายาท (heir)	17	14	9	9	8
ผู้ตาย (dead person)	4	4	3	1	2
พยาน (witness)	3	3	3	2	1
พินัยกรรม (wills)	9	6	9	2	2
เจ้าหนี้ (creditor)	7	4	3	1	2
กำจัดมิให้รับมรดก (exclusion for the succession)	6	4	2	2	6
สละมรดก (renunciation of an estate)	3	2	2	1	3
พินัยกรรมไม่สมบูรณ์ (incomplete will)	4	3	2	2	1
ชำระหนี้ (performing loan)	3	2	1	1	1
ตั้งผู้จัดการมรดก (appointment of the administrator of an estate)	5	3	3	2	1

Table 4. The comparison of performance in term of Diversity measurement found by algorithms

	Diversity Measurements			
keyword	SCRO II	SCRO I	Baseline	Deka
มรดก (estate)	0.74	0.42	0.42	0.53
ผู้จัดการมรดก (administrator of an estate)	0.53	0.53	0.32	0.32
ทายาท (heir)	0.82	0.53	0.53	0.47
ผู้ตาย (dead person)	1.00	0.75	0.25	0.50
พยาน (witness)	1.00	1.00	0.67	0.33
พินัยกรรม (wills)	0.67	1.00	0.22	0.22
เจ้าหนี้ (creditor)	0.57	0.43	0.14	0.29
กำจัดมิให้รับมรดก (exclusion for the succession)	0.67	0.33	0.33	1.00
สละมรดก (renunciation of an estate)	0.67	0.67	0.33	1.00
พินัยกรรมไม่สมบูรณ์ (incomplete will)	0.75	0.50	0.50	0.25
ชำระหนี้ (performing loan)	0.67	0.33	0.33	0.33
ตั้งผู้จัดการมรดก (appointment of the administrator of an estate)	0.60	0.60	0.40	0.20
Average	0.72	0.59	0.37	0.45

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

C. Experimental Results

Considering table 3, we found that SCRO_II can retrieve a set of legal documents with more diversity than those of SCRO_I. The most difficult keyword for algorithms are "estate" and "heir" because they serve as a general term of the domain, so retrieving a set of diversified document for such keyword is considered to be hard. For the keyword, "estate", SCRO_II is able to retrieve a set of documents that have 14 issues whereas SCRO_I gets only 8 issues. For the keyword, "heir", SCRO_II gets 14 different issues from a set of retrieved documents whereas SCRO_I gets 9 issues.

Considering Table 4, the performance measured on diversity shows that the average performance of SCRO_II is higher that that of SCRO_I. SCRO_II gets 0.72, whereas SCRO_I gets 0.59.

SCRO_II and SCRO_I can completely retrieve all issues related to the keyword, ผู้ตาย (dead person) and พยาน (witness). Moreover, SCRO_II and SCRO_I have the same performance on keyword, ผู้จัดการมรดก (administrator of an estate) and สละมรดก (renunciation of an estate).

Considering the baseline algorithms, we surprising found that the Deka system using SQL statement outperforms the traditional information retrieval system using tfidf formula. The reason is that Deka system simply retrieves judges sentences having the keyword and sorts by the document ID no. It is assumed that the succession law cases submitted to the Supreme Court are likely to have different issues. However, the Deka system has less diversity value than SCRO_II because the criteria to determine the performance of any information retrieval system concerns about the ranking process. We want the system to deliver the suitable number of choices (documents) to users so that they can effectively view their search results. Therefore, the objective of our research is to develop an algorithm to effectively retrieve a suitable number of judge sentences that guarantee the variety of issues.

V. Conclusion

This study proposes an ontological approach for Supreme Court Sentence Retrieval. The objective of using ontology is to provide a set of diversify documents for user. We found that using hierarchical style of ontology can provide more variety of law issues in a set of retrieval results. The experiments demonstrate that SCRO_II outperforms SCRO_I algorithm in term of diversity value. Users are satisfied with the judge documents retrieved by SCRO_II.

Acknowledgement

This research was partly supported by Faculty of Science and the graduate school of Kasetsart University.

References

[1] T. Tantisripreecha and N. Soonthornphisaj, "Query Expansion Algorithm for Supreme Court Sentences Retrieval using Ontology," in Proc. of the 48th kasetsart annual conference, 2009.

[2] A. Valente and J. Breuker, " Making ends meet: conceptual models and ontologies in legal problem solving,"In Proc. Of the XI Brazillian AI Symposium, 1994, pp. 1-15.

[3] W. Hohfeld, *Fundamentallegal conceptions as applied in legal reasoning.* YaleUniversity Press, London.

[4] L. Guiraude and Sylvie, D. "Updating ontology in the legal domain", in Proc of the 10th international conference on Artificial intelligence and law, 2005.

[5] M. Saravanan, B. Ravindran, and S. Raman, "Improving legal information retrieval using an ontological framework" In Springer Science Business Media B.V, 2009.

[6] V. R. Benjamins, J. Contreras, P. Casanovas, M.Ayuso, Becue, M., L. Lemus, and C. Urios, (2006) Ontologies of profesional legal knowledge as the basic for intelligent IT support for judges. Artificial Intelligence and Law 12: Page(s) 359-378.

[7] Ahmad Kayed. (2005) Building e-Laws Ontology: New Approach. OTM Workshops. 2005, pp. 826–835.

[8] H. Soonhee, J. Youngim, Y. Aesun, and H. C.Kwon,"Building Korean Classifier Ontology Based on Korean WordNet", 2006, pp. 261-268.

[9] A. Jovic, M. Prcela and D. Gamberger, "Ontologies in Medical Knowledge Representation" in Proc. of the 29th Int. Conf. on Information Technology Interfaces, 2007, pp. 535-540.

[10] N. Henze, P.Dolog, and W. Nejdl, (2004) "Reasoning and Ontologies for Personalized E-Learning in the Semantic Web",Educational Technology & Society, 7 (4), pp.82-97.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)