Using Formal Concept Analysis to Leverage Ontology-Based Yoga Knowledge System

Y. C, LIN
Information Management Department
National Yunlin University of Science & Technology
123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C.
Taiwan
g9323802@yuntech.edu.tw

Abstract: - Over the past decade, the most significant evolution in organization might be the dawn of the new economy in connection with the value of intellectual assets. In this paper, we combine Formal Concept Analysis (FCA) with Protégé in order to construct ontology-based Yoga method serving both as an example, and a knowledge-sharing platform. This may build a picture of experts' knowledge and understand with the general public, allow them to participate in, and be totally aware of, satisfying their requirements. Moreover, the platform will foster to eliminate the phenomenon of information disparity between patients and physicians.

Key-Words: - Formal Concept Analysis, Yoga, Information Retrieval, Ontology, Ontology-based system

1 Introduction

There is not a shred of evidence nowadays people are surrounding with pressure, noise, tension, which turn into lots of problems, such as insomnia, aging, and mental illness. Because of these problems, people start finding ways to improve the quality of life. Yoga, an age-old traditional Indian psycho-philosophical-cultural method, apparently becomes an effective tool of leading people's life to alleviate stress, induce relaxation and provide multiple health benefits. Basically, people can learn Yoga through tool books, magazines, television programs, and Yoga classes. Previously, a certain lack of information was due to insufficient time, manpower, or because information-sharing through education or further explanation may takes long time. Due to a lack of resources and expert knowledge, people are usually ill-informed regarding their particular situation. In this sense, thus, the call to bridge the understanding gap between Yoga learners and practitioners become clear and loud.

Given the lightning fast progress of web service technology associated with the continued rapid growth in data management [16], it is worth to know that we reduce this information gap between learners and practitioners. The users improve Yoga domain knowledge spontaneously when they interact with the Yoga knowledge sharing platform and ask questions.

Finding a best way to construct a user-friendly information retrieval system is a main issue of the system-building and user-interface field and it need to be concerned constantly. However, until recently, the ways we usually use to acquire information still rife with the following problems:

a. The user has to figure out the exact keywords and subjects he/she would like to retrieval.
b. The scope of retrieval is limited: similar concepts by different semantics are hard to detect via a keyword search.
c. The accuracy of the retrieved results is difficult to assess.

A more serious restriction is seen as the present range of functionalities seems to support the user reasonably well, in case where he/she is interested in finding documents described in certain terms or categories. However, the traditional techniques may be of little help if the user wants to reveal the contents of specific sections of the Digital Library, or to mine the concepts contained in a set of articles which have been filtered out by means of some other criterion [11].

How information retrieval (IR) should be set to interact with the user's mindset more properly, and make the retrieval of information easier in executing the functions includes in the system. The main objective of this study is to integrate Formal Concept Analysis (FCA) with Protégé to build a picture of the main structure of knowledge, Yoga served as an example, in terms of ontology to break down the restrictions of IR effectively, and exploit the intention of the document by automatically building a support structure and inter-document relationship.

2 Ontology

Ontology is a formal model of a shared understanding within a certain domain [16]. In accordance to this concept, ontology is treated as a key technique for information management, because of its promise of bringing a consensus to the way in which a particular area of expertise is described. The terminology and the way concepts and objects are organized within the domain restrict the fundamental sharing concept [6].

To declare the ontology, it could be interpreted in two ways [7]:

a. From viewpoints of philosophy, ontology represented a theory of existence. It described what existed in the world and how to configure the world by categories and entities to exhibit things and their relationships.
b. From viewpoints of science, ontology was defined as a set of concepts and attributes that used as construct elements of process system.

The differences between them were purpose and scope [23]. On purpose, the philosophical ontology was to reveal the nature of reality and the computational ontology was to other pragmatic objectives in a specific application. On scope, the philosophical ontology covered with all reality in the entire universe and the computational ontology focused on only the reality of interest application and in only a limited universe. Moreover, the computational ontology was especially focused on a computable and programmable ontology in computer. The term 'ontology' was used to mean as computational ontology to represent a system process in this study.

Ontology was used to provide a presentation of a particular domain. Roughly speaking, a completed ontology was specified by domain ontology and task ontology to represent the static knowledge and problem-solving process in a specific domain [14]. Both of them had been employed in this study and been introduced in section 2.1 and 2.2.

2.1 Domain ontology

Domain ontology was considered to be a thesaurus that composed of numerous domain concepts and their associations [24]. It employed concepts and associations to represent all the information about static knowledge in a particular domain. In domain ontology, every concept was described by the frame of concepts and sub-concepts and was associated by the relationships of relevance, inheritance and belongs to. To declare it, the relationships of relevance was concepts in a common concept where possessed the same attributes, the inheritance was an underclass concept that inherited the most attributes in its hierarchy and the belongs to was a concept within one or several concepts.

For the simplification of system description, the description logic of domain ontology was following with frame-based logic of protégé by class, slot and facet in this study. The class, slot and facet were meant as knowledge concept, attribute and the value of attribute respectively.

Sum up above, domain ontology should be a number of classes that expressed by slots and facets and be a lot of clarified associations which indicated by relevance, inheritance and belongs to to construct a static knowledge within a bounded domain. Furthermore, the description depth of domain ontology was relying on the demand of application as mentioned in section 2. An exemplary case of domain ontology was shown as Fig.1, where the relationships of belongs to was noted as 'is-a' and the inheritance was drawn as real line.

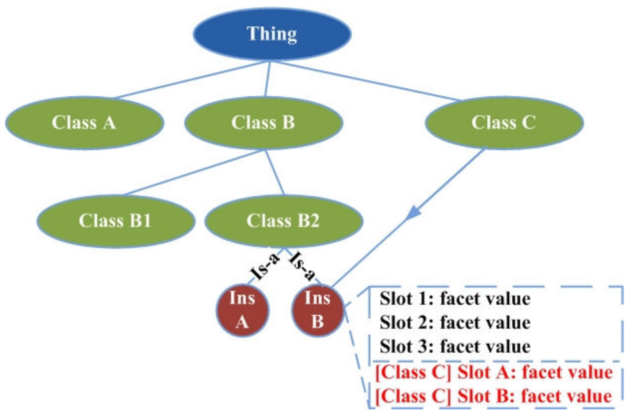

Fig.1 Exemplary case of domain ontology

Domain ontology was constructed of the complex tree-structure that out of balance nodes or fixed shape. And the complexity of tree-structure was only depended on the contents of domain knowledge. In the exemplary case, the relationships between instance A and B was relevance. Both of them were belonged to a common class named B2 class. The slots of B2 class were slot 1, slot 2 and slot3 and it also inherited the slots from C class and therefore it proposed 5 slots for describing the knowledge concept of B2 class.

By objective process of domain ontology, the knowledge became more convenient to share and reuse. Currently, domain ontology had been successfully used for knowledge description in different domains [2][3][4][5][8][9]. However, the domain ontology contained only the static knowledge of a particular domain and the lacked of universality was made up by task ontology.

2.2 Task ontology

Task ontology, also known as method ontology, was a set of operand and operator for characterizing the problem solving process in generic task [1]. A task was considered as a sequence of problem solving steps [12]. The operand of task ontology was corresponded to the terminology of domain ontology and the operator was specified for the necessary of computing. The relationships between domain and task ontology were shown as Fig. 2.

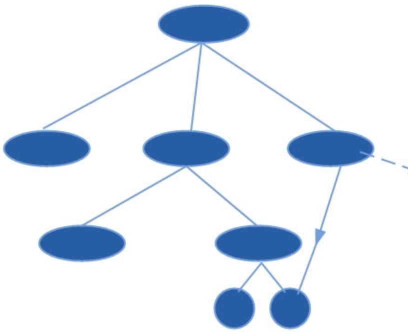

Fig.2 The relationships between domain and task ontology

Task ontology simulated the problem solving processes at the conceptual level by its descriptiveness and readability to represent the execution process in terms of conceptual level primitives. By task ontology, knowledge constructor was able to describe their own task easily and to make the task description runnable [22].

Task ontology represented as system of semantic features to define key task concepts by a rule based reasoning approach. If/else was one of the most common formalization in task ontology. After introducing the dynamic knowledge of task ontology, the ontology of a particular domain became an inducible process and more knowledge were able to be reused. Nowadays, task ontology became an available methodology in dynamic knowledge descriptions [18][19][20][21][22]. Task ontology offered an effective methodology and vocabulary for disassembling the solving-process and synthesizing the knowledge-based systems.

In summary, domain ontology and task ontology were indispensable to a knowledge system. The knowledge of Yoga is represented as ontology concepts (instance and associations of static knowledge) and generic tasks (rule sets of problem-solving process) by ontology approach in this study. And therefore, this knowledge system will become more definite, complete, consistent and convenient to share, store and reuse by domain and task knowledge modeled in the form of ontology.

2.3 Using ontology in knowledge retrieval

The principle of using ontology in a data retrieval domain was described by Andreas and Ludger[10]. In the general case of a blank or small answer set, taxonomic information is used for extending the question by the use of sub-concepts or super-concepts. The more precisely a domain is modeled, the better typical questions and answers are understood, and similar ideas can then be better applied to other, non-taxonomic relationships. When a situation is changed for a complex learner’s demand, the retrieval of potentially useful documented experiences is supported by a detailed Yoga set in terms of ‘is-a’ and ‘has-part’ relations. Similarly, potentially useful situations can also be assessed using modeled links describing breathing and relaxation connections between postures. As we know, the more specific the description a domain, the more powerful the inferences for questions that will be displayed, and the more reformulation and information network expansion that will be possible [13].

Ontology applied in data retrieval definitely changes the strategy from the pull to push style [10]. Previously, the type of pull style data retrieval which means the way the retrievers inserted a certain keyword in the system to lookup information. However, nowadays ontology conceptual construction provides the vocabulary for expressing personal interest profiles for information, while the push service automatically delivers the data and information for categories in which a user is interested.

This study applies Formal Concept Analysis, which is a mathematical principle to the main spirit of ontology, allowing the knowledge to be constructed more sensible and reasonable.

2.4 Formal concept analysis

Formal concept analysis was originally developed as a field of applied mathematics based on a mathematical concept and concept hierarchy at the beginning. But after more than a decade of development, it has been applied to various different fields, such as: psychology, sociology, anthropology, medicine, biology, linguistics, computer sciences, math etc [17]. In the perspective of philosophy, a concept is a unit of thoughts; an idea for reinforcing connections with our general culture by interpreting the theory as concretely as possible.

FCA has been successfully drawn on various real-world projects. There are some questions we should ask in order that the power and success of FCA can be explained: what is the unique peculiarity that makes FAC so applicable and what are the advantages of FCA compared to the usual form of lattice theory? It is not simply a mathematically structured theory; the answer lies in the underlying philosophy of FCA. The main idea came from Hentig, who explained that the main concept of FCA is the restructuring of science disciplines. Sciences have to examine their disciplines, which means uncovering their hidden purposes, to declare their real purpose, to select and adjust their means according to those purposes, to explain possible consequences comprehensively and publicly, and to make their methods of scientific findings and results accessible via everyday language [15][17].

Adopting FCA can bring us some advantage: the main ideas are quickly understood by means of formal lattices, and the knowledge model follows FCA concepts which explain the extraction schema from the documents or experts opinions we have. As a consequence, this study applies FCA as an automatic technique to elicit the attributes of dependency schemas extracted from the documents and Yoga references. Therefore, building up ontology as a set of user thoughts consists of two parts: the extension and the intention. The extension covers all objects which come from the concept, while the intention comprises all of the attributes. Later on, we will introduce the Formal Concept Analysis tools, which we use to analyze the connection between the objects and attributes.

2.5 FCA application for Information retrieval

Putting FCA into practice for IR usually involves the three steps listed below: (1) extracting a set of index terms that describe each document of the collection, (2) relating the document terms generated from the previous step to each other via the concept lattice, (3) visualization of the concept lattice, constructed by step (2), of the structure diagram. The outcome of each step may have a profound effect on the overall application [11].

Step1: Extracting indexes from documents

The IR scholars brought up the principle to extract a set of index terms from the documents. Those terms describe the documents appropriately. In order to maximize the performance of a certain domain IR task, researchers can use selective criterion to restrict the index set, or separate a needed index from the inordinate one by means of weights in the documents.

Step2: Lattice Construction

Concept eliciting is always regarded as the key factor in the success of Formal Concept Analysis study. However, the more objects/documents we get, the more concepts that they contain. It is well known that the size of a concept lattice may grow exponentially along with the number of objects. It is always debated what is the best way to control the number of concepts extracted from documents by several field experts and the different backgrounds of applied researchers. However, the Claudio and Giovanni [11] study referred to keyword limitations to manipulate the increasing number of extracted concepts.

It generally consists of a focus concept technique, namely, to select a focused-concept/centered-concept, then observe the concepts nearby and choose which is closest to the centered-concept that can be merged with. For instance, in order to make a more objective decision, sometimes it might be assisted by a point-and-click graphical user interface showing the condition of concepts and its neighbors, or selected by the user through a simple minded string matching, in order to make a more objective decision. The contribution made by all of the nearest neighbors of a given concept, has been addressed, both for building a full lattice finding the portion of lattice around which concept is centered. The example code below seeks to demonstrate the processing principle associated with the neighboring concepts by virtual codes.

The input of this function which finds the neighboring concept input is Context (G,M,I), concept (X,Y) of context (G,M,I), and the output is the set of lower neighbors of (X,Y) in the concept lattice of (G,M,I). A formal context is defined as a triplet (G,M,I) where there are two sets: G (represents objects) and M (represents attributes), and a binary incidence relation $I \subseteq G \times M$, expressing which attributes describe each object (or which objects are described by using an attribute), i.e. the algorithm for determining the set of lower neighbors of a given concept is derived by measuring the distance between different concepts, and merging two concepts together if they are close enough.

The concept code for finding the neighboring concept

```
Lower Neighbors := NULL
TestedCard := NULL
For each m ⊂ M\Y \*M is one of attributes of
concept Y
```

X1:= X ⊂ {m}*declare new variable and make it obey {m}
Y1:=X'1
If (X1,Y1)∉ TestedCard
*Checking concepts(X1,Y1) in list or not
Then
Add (X1,Y1) to TestedCard *add a new concept in list
Count (X1,Y1) :=1
Else
Count (X1,Y1) := Count (X1,Y1)+1
If (|X1|-|Y1|) = Count (X1,Y1)
Then *compute the distance between two concepts
Add (X1,Y1) to lower Neighbors

Step3: Effective Lattice Visualization

In some cases there are some problems with showing the lattice: The outcome concept lattices of a real application are so huge that it can only show partially on the screen or graph.

Showing or hiding parts of the lattice via an interactive specification of a focus concept and/or subsets of terms is the common solution of this problem. Several simple and useful approaches are used to represent the concepts and their inter-relationships. For example, nested line diagrams, similar to the tree structure, make the focus concepts the roots and associate each sequence of concepts below the focus with a path like the trunk of a tree. However, there are still some differences between the tree and FCA nested line diagrams.

The FCA nested line diagrams are based on combining multiple partial views of the data represented in the context. There are three steps to build an FCA nested line diagram:

a. Two or more subsets of attributes are chosen by the user.
b. The concept lattices of the sub-context are identified by the attribute subsets of the previous steps chosen by the users.
c. The full concept lattice is embedded in the direct product of the lattices of sub-context as a joint-semi-lattice.

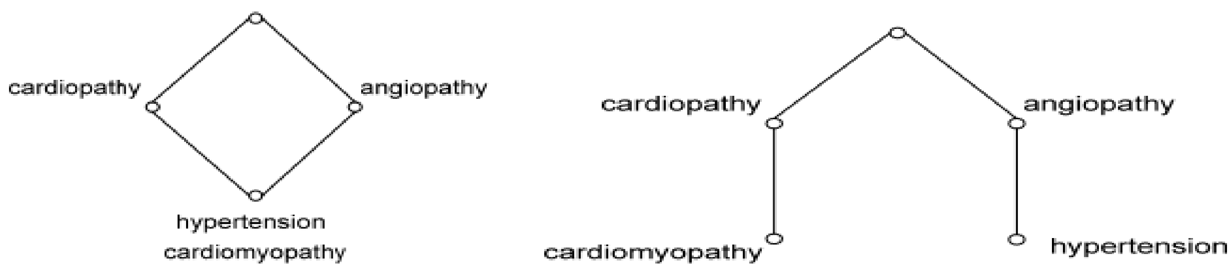

Fig. 3 Nested line diagram and tree structure diagram

In general, the effect of building nested line diagrams is mapping concepts into a clear identifiable picture of reality, which means having several complete lattices of a partial context nested into one another rather than a partial lattice of a context.

The nested line diagrams which are developed from FCA concepts can sometimes make a better logical coverage of contents than the standard tree structures. For instance, the Fig.3 shows hypertension is a type of cardiovascular disease, but cardiopathy is the particular type of disease which has a high relationship with hypertension. Regarding the disadvantages of class misunderstandings which we mentioned before, the FCA nested line diagrams not only avoid the problem of classing, but also have several advantages, such as the symbol of hierarchical folders used for storing and retrieving the correct information from an information library, book-marks, and system menu items generated automatically, etc., from the user's point of view.

The nested line diagrams represent the concepts elicited from the original documents, and they are more user-friendly, enabling users familiar with the system to work more efficiently. Therefore, nested line diagrams could satisfy the need users without training. It takes less space on the screen than the traditional ways, which is, drawing the concepts of domain information more completely.

Another advantage of the nested line diagrams is greater flexibility in profiling the facts within the certain scope of a domain. Drawing the full lattices of each sub-context which are merely a part of the larger knowledge picture is possible.

This creates the effect that sub-contexts are smaller and easy to understand. In this regard, control of the size of each local diagram for different situations of retrieval should not exceed the number of possible combinations of the attribute values present in the corresponding sub-context, regardless of the number of objects in the database.

It is noticeable that building nested line diagrams speedily may revolutionize the application of graphical interfacing, no matter what the webpage information display or database retrieval. Moreover, it may encourage an interest in the techniques for lattice visualization, in addition to exploring the use of proposed alternative visual layouts in the visual information field.

3 Implementation of Yoga

We will introduce the Yoga Health Knowledge Management System (YHKMS) with more detail in this section. The system is constructed by the three step principle mentioned earlier. We use the integrated software tool, Protégé, to construct the system retrieval and organize queries about the articles or the principles related to domain knowledge. As we mentioned above, ontology can be seen as defining a set of concepts and their inter-relationships. Such relationships allow experts to embed their knowledge to be accessed more easily, and encourage the creation of a knowledge-acquisition tool which is designed to be domain-specific. The resulting ontology concepts can then be used along a problem-solving method to answer the questions. The application of Protégé [25] is the end product created for solving an end-user problem with the ontology-based data. It employs the appropriate problem-solving, expert-system, or decision-support method. Moreover, there are four strong points which can be achieved by Protégé: □

a. The modeling of ontology of classes describing a particular subject.

b. The creation of a knowledge-acquisition tool for collecting information.

c. The entering of specific instances of data and the creation of a data base.

The execution of applications, following the completion of these characteristics, was the reason we adopted Protégé to construct the framework of a Search Data Management subsystem (SDM).

SDM is the core of an YHKMS information retrieval platform, which built with Yoga ontology by Protégé. This design comes with some advantage: the capability of handling ambiguous questions from different users. SDM provides two kinds of retrieval methods for acquiring certain information for the users. First, an intelligent query gives the user additional choices between the expert and the beginner. We presume that the beginner's behavior will not be the same as an expert. If a beginner looks up the Yoga domain data from YHKMS, multiple combo-boxes supply the user with a flexible combination for his/her understanding and purpose.

Generally speaking, experts' queries require more accuracy and efficiency. The purpose of the other retrieval mechanism is to provide a keyword method for retrieving information, which may be more efficient than the first way we mentioned, and even more satisfying the needs of experts. There is an ancillary function: when searching for specific news and yoga knowledge, users can insert the keyword they interested in, and the function will return the relevant yoga news report and articles as feedback as showed in Fig.4.

In the following section, we will introduce the three main steps for building the searching function, as well as the process for constructing the ontology by Protégé.

Fig.4 Interface of Yoga Health Knowledge System Webpage

3.1 Extracting a set of index terms

As we have discussed above, although there are many techniques can be applied to improve the effectiveness of retrieved knowledge, the data still plays an important role in the information system. However, a good data sharing platform with the latest technology can be useless if there is no file in it.

Emphasizing the source of the information itself is always the best strategy when building an information retrieval system. In order to construct platform, we collect any kind of Yoga documents and references. We define the system as a Yoga consulting platform. We control the scope of the data applied in Yoga knowledge domain. The system will then try to figure out the relationship of the Yoga terminology mentioned in the references and documents.

In this step, we focus on creating the slots for each information item by using the Protégé application which is suitable for designing ontology-based systems. These slots are also known as 'roles' in the description logic and relations in FCA construction and other object-oriented notions. They can be called attributes in some other formats.

This study utilizes data from Yoga-related domain, such as literature, documents, CDs, and so forth. We create Yoga ontology via the Protégé integrated software. The process of setup interfaces is shown as below.

For example, the effect of practicing Yoga can be divided into several parts: preventing disease, curing disease, and maintaining beauty. These effects could be the super-classes, and then we place the special regions which are treated as sub-classes into appropriate super-classes. Preventing dandruff and preventing tinnitus are classified into the same super-class. At the lowest level, the scope becomes narrow and more specific, and we find the class where the slots for Yoga are set. Slots play the role of description logics, and they are relative to FCA construction. Some FCA research label attributes as slots, which is the characteristic part of an object. Different acts of Yoga provided with different character may correspond with different attributes. Those different places are specified by different allotted-values of slots, as displayed in Fig.5.

The different slots indicate to different results. For example, the slot preventing disease points to the effect caused by Yoga and which act should be done; the slot degree of movement represents how much strength should be applied to during the movement. The slot of effective positions, is the style of Yoga stating the manner in how to do the act correctly, such as stretch the arms smoothly, raise the upper part of the body gently like a snake, and so on. Every act of yoga is a setup slot in Protégé for modelling ontology structure. Yoga is described by different slots in every aspect. Furthermore, building the relationship and constructing concept lattices must be based on these slots which belong to each act of Yoga, as in the following steps.

3.2 Concept lattice construction

Ontology illustrates the concepts and the relationship between each other in certain domains. The main task in this section is mapping the concepts into the appropriate classes. Classes are interpreted as sets that contain instances. In order to find the common parts of two classes and determine their relationship, the construction of links between the concepts ranks with each other according to related degree. In order to present the concept more completely, several similar objects are merged as a new concept in representing them.

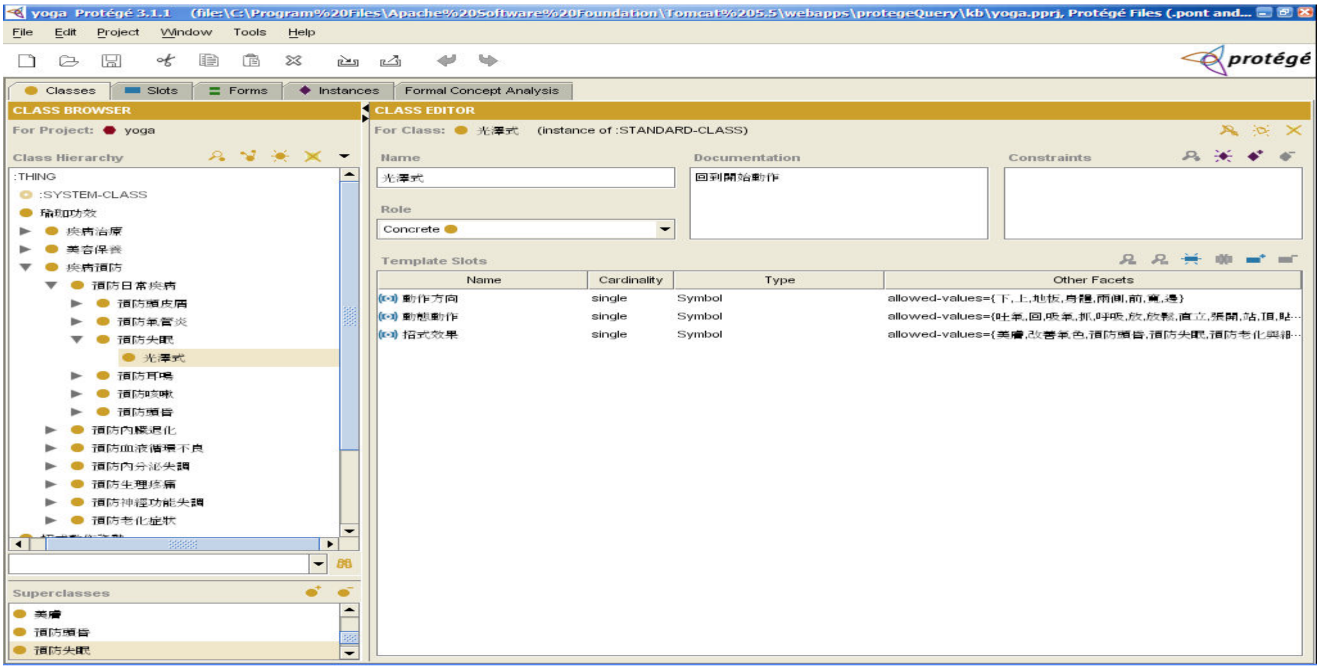

Fig.5 Interface of constructing Protégé classes

As the diagram shows below, because of the Yoga ontology we built is both large and complex, it is necessary to set the concepts for representing the sub-class. In this process, five concepts gradually emerge and become the mallow of the Yoga ontology: “The effect of practices”, “The direction of movements”, “The time of movements”, “The postures”, and “The degree of movements”.
Although the diagram below cannot be seen clearly, it shows the result which was built via formal concept analysis. There are something we can do to reassure people or for a beginner to understand. For example, setting the attributes and concepts through the use of mathematics, and making the facts more objective and reasonable. Furthermore, it makes the constructed data-base cheaper and diminishes the need for experts.

3.3 Queries Outcome Visualization

Users can look up what they are most interested in from certain Yoga domain information via the relationship between the concepts constructed in step 2 as shown in Fig.6. And, the Fig.7 displays below is represented the nest line diagram. Although this diagram doesn’t show the complete yoga domain, when selected by a user it will provide them with adequate general information concerning certain Yoga posture.

Fig.6 Yoga knowledge Ontology

Protégé compiled a domain information rule and relationship by the XML syntax, which contains the rule of the Yoga postures. Indeed, linking the postures, treatments and effects with each other, and constructing different parts of the experts’ know-how into a knowledge-triangle, can give consideration to two or more aspects of health care. As Fig.8 shows, the XML format is compiled automatically by the Protégé integrated software.
This may avoid the possible error caused by user misunderstanding, because users might misread the contents of the system’s response. Also, the user may personally judge, through feedback, whether or not his/her understanding of Yoga knowledge corresponds with that of the experts according to the whole picture awareness of the domain.

4 Conclusion

In order to bridge the understanding gap between the Yoga learners and practitioners, we made efforts in improving the knowledge management as applied to the Yoga domain. Serving Yoga postures as an example, we build an automatic ontology-based system menu with technique of FCA, and replace the pull style with a push style of IR service. There are three features of the system developed by our study:

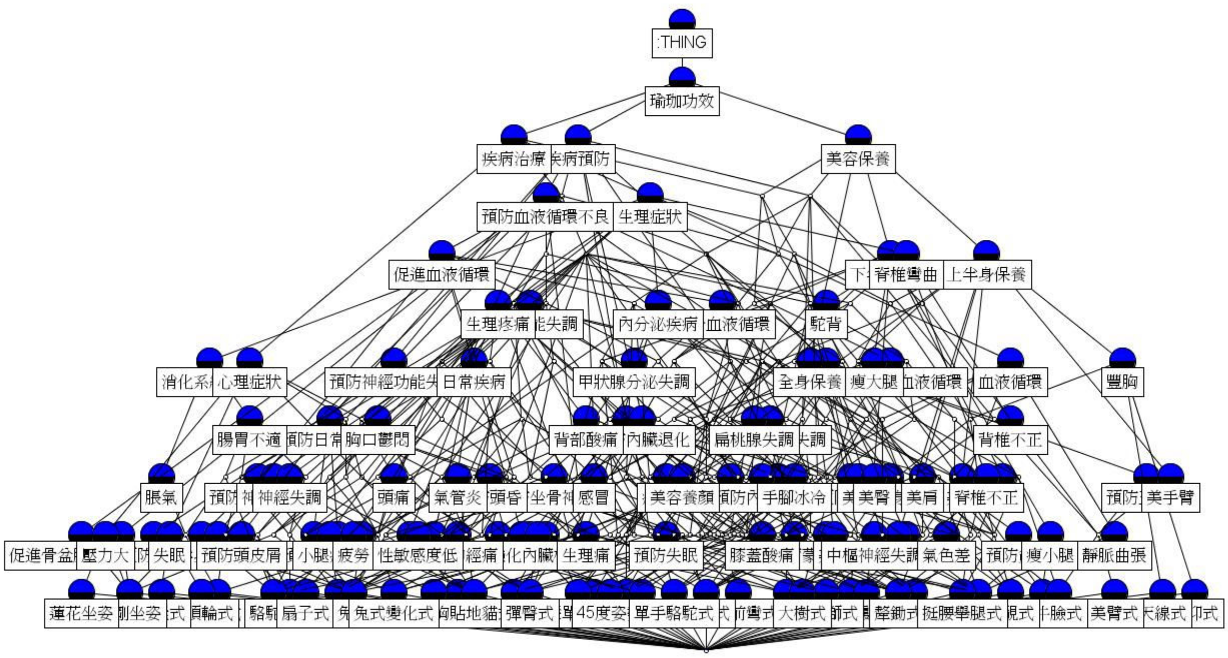

Fig.8 The XML syntax of Yoga posture

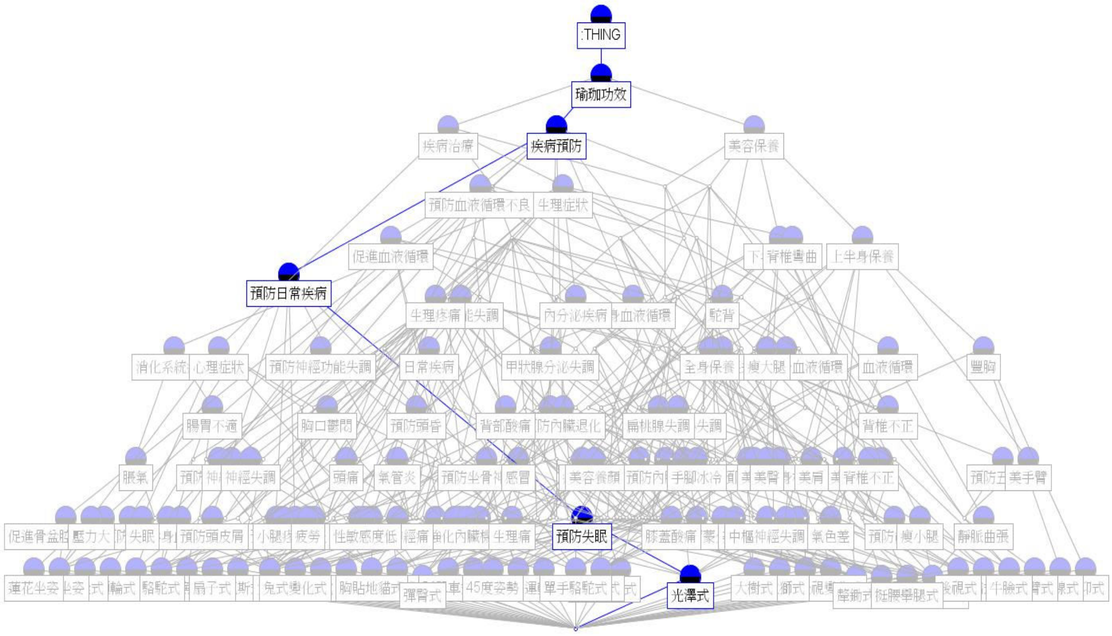

Fig.7 Nest line of Yoga knowledge


```
- <class>
    <name>光澤式</name>
    <type>:STANDARD-CLASS</type>
  + <own_slot_value>
  - <own_slot_value>
      <slot_reference>:DOCUMENTATION</slot_reference>
      <value value_type="string">回到開始動作</value>
    </own_slot_value>
    <superclass>往下方</superclass>
    <superclass>往上方</superclass>
    <superclass>往兩側</superclass>
    <superclass>往前方</superclass>
    <superclass>身體停留</superclass>
    <superclass>吐氣</superclass>
    <superclass>吸氣</superclass>
    <superclass>手抓住</superclass>
    <superclass>身體放鬆</superclass>
    <superclass>站立</superclass>
    <superclass>身體頂地</superclass>
    <superclass>調整呼吸</superclass>
    <superclass>氣色差</superclass>
    <superclass>美膚</superclass>
    <superclass>預防頭昏</superclass>
    <superclass>預防失眠</superclass>
    <superclass>預防老化</superclass>
    <superclass>預防細紋</superclass>
  + <template_facet_value>
  + <template_facet_value>
  + <template_facet_value>
  </class>
```


a. Automatically matches the effects inserted by the users, with the postures.
b. Matching the effects with steps of postures according to the user' s request and requests will show how Yoga postures relieve pain.
c. Generating a hierarchical information retrieval mechanism of Yoga information to reduce obstacles in system accessing.

Our ultimate goal is improving the traditional ways of managing both information items, such as documents, and medical science literature, and domain data of Yoga, as well as minimizing the additional effort required in information retrieval and content management. To achieve this goal, we develop FCA-constructed retrieval mechanism, and emphasize an intelligent query interface when building the hierarchy menu on the ontology concept. The concept greatly differs from previous ones, as it formulates user queries accordingly; interpreting both the question and answers can avoid ambiguity. Such normalization and retrieval promise a precise search beyond what is possible with the current keyword-based methods. Although unaware of the semantics of terms, it is easier for the user to control; the application we developed makes it possible to create an automatic system menu for helping the user to effectively discover and question the domain while the complexity of the ontology-based information retrieval system is hidden.

Furthermore, creating a platform for sharing the Yoga data with users and eliminating the phenomenon of the information disparity between the learners and practitioners are the long-term object of this study. This study's perspective creates "images" for eliminating the disparity, and offers the potential for generating new insights into the field as it advances.

References:

[1] D. Rajpathak, E. Motta, Z. Zdrahal, & R. Roy, A Generic Library of Problem Solving Methods for Scheduling Applications, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 6, 2006, pp.815-828.

[2] J. Charlet, B. Bachimont & M. C. Jaulent, , Building medical ontologies by terminology extraction from texts: an experiment for the intensive care units, *Computer in Biology and Medicine*, Vol. 36, No. 7-8, 2006, pp. 857-870.

[3] L. V. Elst & A. Abecker, Ontologies for information management: balancing formality, stability, and sharing scope, *Expert Systems with Applications*, Vol. 23, No. 4, 2002, pp. 357-366.

[4] M. Beveridge & J. Fox, Automatic generation of spoken dialogue from medical plans and ontologies, *Journal of Biomedical informatics,* Vol. 39, No. 5, 2006, pp. 482-499.

[5] Q. Tho, S. Hui & A. Fong, A citation-based document retrieval system for finding research expertise, *Information Processing and Management*, Vol. 43, No. 1, 2007,pp.248-264.

[6] R. Kishore, R. Sharman & R. Ramesh, Computational ontologies and information systems: I. foundations, *Communications of the Association for Information Systems*, Vol. 14, 2004, pp. 158-183.

[7] R. Mizoguchi, T. Sano & Y. Kitamura, An Ontology-Based Human Friendly Message Generation in a Multiagent Human Media System for Oil Refinery Plant Operation, *IEEE Systems, Man and Cybernetics Society*, 1999, pp. 648-653.

[8] S. S. Weng, H. J. Tsai, S. C. Liu & C. H. Hsu, Ontology construction for information classification, *Expert Systems with Applications*, Vol. 31, No. 1, 2006, pp. 1-12.

[9] T. Asikainen, T. Männistö & Soininen T. K., A Domain Ontology for Modelling Variability in Software Product Families, *Advanced Engineering Informatics*, Vol. 21, No. 1, 2007, pp. 23-40.

[10] A. Andreas & V. E. Ludger, *Ontologies for knowledge management*, Hand book of Ontology part 4, Springer Publishing, 2004.

[11] C. Claudio & R. Giovanni, *Using Concepts Lattices for Text Retrieval and Mining, Formal Concept Analysis Foundations and Applications*, Springer Publishing, 2005.

[12] N. J. I. Mars, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, Amsterdam: IOS, 1995.

[13] P. Mika, V. Iosif, Y. Sure& H. Akkeermans, *Ontology-based content management in a virtual organization*, Hand book of Ontology part 4, Springer Publishing, 2003.

[14] R. M. Botting, *A formal and structured approach to the use of task analysis in accident modeling*, Academic Press,49,pp223-244,1998.

[15] R. Mizoguchi, J. Vanwelkenhuysen, M. Ikeda, *Task Ontology for Reuse of Problem Solving Knowledge. Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing*, pp.46-59, IOS Press, 1995.

[16] R. T. Gruber, *Towards Principles for the design of Ontologies Used for Knowledge Sharing, Formal Ontology in Conceptual Analysis and Knowledge Representation, Deventer*, The Netherlands, Kluwer Academic Publishers, 1993.

[17] U. Priss, *Linguistic Applications of Formal Concept Analysis, Formal Concept Analysis Foundations and Applications*, Springer Publishing, 2005.

[18] A. I. Chao, B. C. Krikeles, A. E. Lusignan& E. Starczewski, An extensible, ontology-based, distributed information system architecture , Proceedings of the Sixth International Conference on Information Fusion, 2003, pp. 642-649.

[19] L. Chen, N. Shadbolt, F. Tao, C. Puleston, C. Goble & S. Cox, Exploiting Semantics for e-Science on the Semantic Grid, Proceedings IEEE/WIC International Conference on Web Intelligence/Intelligent Agent Technology Workshop on Knowledge Grid and Grid Intelligence,2003, pp. 13-16.

[20] M .Ikeda, K. Seta, O. Kakusho & R. Mizoguchi, Capturing a Conceptual Model for End-User Programming: Task Ontology As a Static User Model, 6th International Conference on User Modeling, 1997, pp. 203-214.

[21] M .Ikeda, K. Seta, O. Kakusho & R. Mizoguchi, Task ontology: ontology for building conceptual problem solving models, Proceedings of ECAI98 Workshop on

Applications of ontologies and problem-solving model, 1998, pp. 126-133.
[22] M. Ikeda, K. Seta & R. Mizoguchi, Task Ontology Makes It Easier To Use Authoring Tools, Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, 1997, pp. 342-347.
[23] R. Kishore, R. Ramesh & R. Sharman, Computational Ontologies : foundations, representations, and methods , Proceedings of Ninth Americas Conference on Information Systems (9th AMCIS), 2003, pp. 3178-3189.
[24] Z. Y. Li, Z. X. Wang, Y. Y. Yang, Y. Wu, & Y. Liu, Towards a Multiple Ontology Framework for Requirements Elicitation and Reuse, Proceedings of the 31st Annual International Computer Software and Applications Conference, 2007, pp. 189-195.
[25] Protégé User Document, Retrieved on Oct 1, 2009, Available at: http://protege.stanford.edu/doc/users.html