CONSTRUCTING A BAYESIAN BELIEF NETWORK TO GENERATE LEARNING PATH IN ADAPTIVE HYPERMEDIA SYSTEM

NGUYEN VIET ANH, NGUYEN VIET HA, HO SI DAM

College of Technology- Vietnam National University Hanoi, Vietnam; vietanh@vnu.edu.vn

Abstract. There are many methods and techniques which have been promoted to develop adaptive hypermedia systems [1]. Our model approach [2], generating adaptive courses based on learner's profile which learner's includes background, skills, style...etc. One of important steps in our model is to generate learning path adaptive for each learner. In this paper, we promote an algorithm based on shortest path search algorithm to evaluate learning object (LO) based on its attributes [3] and constructed a Bayesian Belief Network (BBN) to generate learning path for each learner.

Tóm tắt. Nhiều phương pháp cũng như kỹ thuật được đề xuất để phát triển các hệ thống học thích nghi. Mô hình của chúng tôi phát triển nhằm tạo ra các khóa học thích nghi dựa trên các thông tin về người học như kiến thức, kỹ năng, sở thích v..v. Một trong những bước quan trọng của mô hình là tạo ra các tiến trình học thích nghi cho từng người học. Bài báo này chúng tôi trình bày thuật toán dựa trên thuật toán tìm đường đi ngắn nhất để lựa chọn các đối tượng học dựa vào thuộc tính của chúng và xây dựng mạng xác suất Bayesian Belief để tạo ra các tiến trình học phù hợp với nhu cầu người học.

INTRODUCTION

With innovation of internet technology, web based training systems have been developed to support learner who can learn every time, everywhere. However, hardly do the learners obtain knowledge that they need because of huge course information. There are many approaches to develop adaptive hypermedia as well as personalized systems to solve problem such as MELOT (http://www.merlot.org), CAREO (http://www.careo.org), and SMETE (http://www.smete.org/smete). They adopt standard e-learning metadata specifications to describe LOs, they use full text queries to access Los in a disconnected way from actual learner's navigation [4]. The new standards for LO metadata (http://itsc.ieee.org/wg12/) are defined in order to classify LO among them, but teachers and developers may still face problems when choosing LO to adapt with learner's demand because LO 's attributes do not have enough information for classifying processes in consideration with learner demands. Considerable work has been conducted on adaptive hypermedia system [4, 5], WebCL (http://www.webcl.net.cn) is considered to be relevant with our approach. However their approach to LOs searching based on keyword matching of learning object content, as well as the LOs sequencing process is quite different from our approach.

In our approach, the ACG system [2] supplement some LO attributes which are utilized to build the course structure or knowledge maps for each learner. To do that, our model includes three modules: learner module, content module and view module. The first module manages learner modeling as well as profile of them. The second module generates suitable learning path for each learner based on learner's profile. The last module represents suitable course outline for each learner. We provide a tool for teachers or designers to develop their course knowledge maps. This is a direct acyclic graph which includes vertex and direct edge, the former represents knowledge unit which is constructed by one or some LO, the later represents knowledge unit relationships. Our goal is generating learning path which is in nature of sub knowledge map for each learners based on their profile. In selecting process whether vertices are selected or not depend on their weight and all of LO weights which are constructed. To solve this problem, in early stage we promoted an algorithm based on shortest path searching to select learning path for each attribute of LO and we construct a Bayesian Belief Network to generate learning path.

In the next section, we focus on semantic model of knowledge unit. The candidate learning path selecting process as well as learning path generating process had described in section 2 and section 3. In section 4, we constructed a BBN to create the learning path for leaner. Finally, in the last section the experimental results are reported. We draw conclusions and indicate future directions of our research.

1. SEMANTIC MODEL OF KNOWLEDGE UNIT

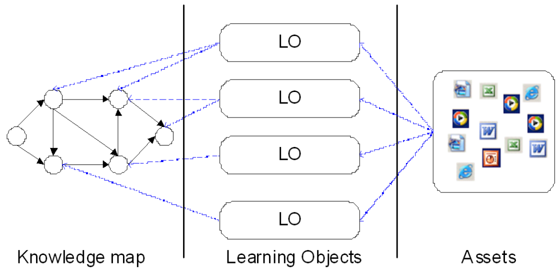


Figure 1. Semantic model of knowledge unit

To provide adaptability, not only does knowledge unit consider SCORM standard but also defines the following attributes:

- Prerequisites: For required object that learner have to visit when browsing the course such as concept
- Master Level: To classify learner level such as beginner, intermediate, advanced and expert
- Difficulty Level: Difficulty reveals that learning objects is easy to learn or not.
- Required Time: corresponding with difficult level, required time reveals the minimum time calculated in minutes which learner need to finish.
- Relation: Show the specific relationship between learning objects and others in the appropriate learning sequence.

- Interactive Style: Show good strategy for approaching learning object such as: top-down, bottom-up, consequence, parallel.
- For Skill: For teaching learner skill such as: understanding, deducing, etc.

Because knowledge unit is constructed by one or more LO, LO also have all attributes of knowledge unit. Besides, we also supplement some attributes for assets. Teachers or course designers will help to assign weight for assets when they create the course. Knowledge unit, LO, and assets take form of object class to inherit their attributes.

2. THE LEARNING PATH GENERATING PROCESS

For each learner who participates in the course, our system will automatically generates the best learning path for learner which is based on learner's profile as well as knowledge map that had been design by teacher or designer for learning syllabus plan. The learning path generating process includes some steps, which are shown in Figure 2.

Step 1. Learner evaluating. Based on learner demands and learner profile, the process evaluating learner in order to classify learners as well as to get demands for the course which learner intend to participate in.

Step 2. Knowledge mapping. This step bases on LOs database, and some result of step one, teachers or course designers outline knowledge map as a graph with vertices represent knowledge unit and edges represent relationship among knowledge unit.

Step 3. Candidate learning path selecting. Base on learner's profile, LO's attributes, in this step we select some candidate learning paths which are learning paths for learner when all knowledge unit in the graph is focused on only one attribute. For example, if learner demand focuses on require time, difficulty level attributes; there are two candidate paths corresponding with two attributes which are mentioned above.

Step 4. Learning path generating. In step 3, we had created some candidate learning path for learner. To construct a learning path which meets learner's demand at maximum is based on probability of knowledge unit in candidate learning paths; we constructed a Bayesian Belief Network to resolve it.

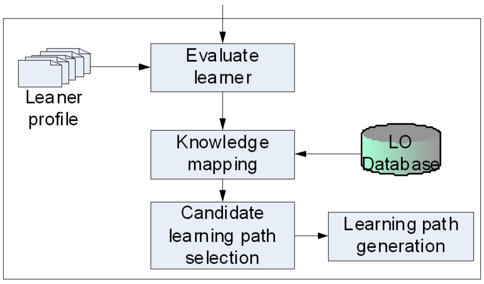


Figure 2. Learning path generation process

Two first steps of process, we deeply described in [1, 2], in this paper we give details of the candidate learning path as well as learning path generating process in the next sections.

3. THE CANDIDATE LEARNING PATH SELECTING PROCESS

3.1. Candidate learning path

Definition 1. The knowledge map is a direct graph $G = (V, E)$ with $V = \{v_0, v_1, ..., v_n)$ is set of vertices, v_i represent knowledge unit, $E = \{e_0, e_1, ..., e_n\}$ are set of edges, e_i represents relationship among knowledge unit. All of e_i are signed a weight w_i whose value reveals the difficulty to access a vertex coming from a previous one.

Definition 2. The learning path is set of vertices $V = \{v_s, v_i, ..., v_j, v_e\}$ in knowledge map which are knowledge unit that learners need to browse when they participate in their course to finish. V_s is the starting point for learner to reach V_e - the target knowledge unit.

Definition 3. The candidate learning path is a learning path that have $\Sigma w_i \rightarrow \min$ or $\Sigma w_i \rightarrow \max$ $(i = s..e)$ with min or max value in threshold.

3.2. Candidate learning path selecting algorithm

Our target is to generate learning path for each learner which is based on his or her profile. To do this, in the first stage we select learning path in knowledge map corresponding with an attribute of LO, so the number of candidate paths is equal to the number of LO attributes, Each path is candidate path for one attribute of LO, so it is independent with each other. In the next step, we generate learning path based on these candidate paths which are the results in first step. We construct a Bayesian Belief Network (BBN) to resolve it. To select the path in the first step, we promote an algorithm based on the shortest path searching.

```
Input: The knowledge map $G = \{V, E\}$; The $\partial$ is a threshold; $V_s$ the staring knowledge unit; $V_e$ the target knowledge unit
Output: A candidate path
Begin
    $S = \{V_s\}$
    For $i := 2$ to $n$ do
    Begin
        $D[i] := C[1, i]$;
        $P[i] = \{V_s\}$;
    End;
    While $V - S \neq \phi$ do
        Begin
            Select $v \in V - S$ that $D[v] \rightarrow \min$
            $S := S \cup \{v\}$;
            For each $w \in V - S$ do
            If $D[v] + C[v, w] < D[w]$ then
                Begin
                $D[w] := D[v] + C[v, w]$;
                $P[w] := v$;
                End;
        End;
End;
```

With $C[i,j]$ is weight value of e_k - edge that represents relationship between knowledge unit i, and j. if i,j do not have relationship w is ∞. $D[u]$ represents relationship value among $\{V_s\}$ and u. $P[u]$ represents trace of path, with $P[u]=v$ if there is a path $v \to u$.

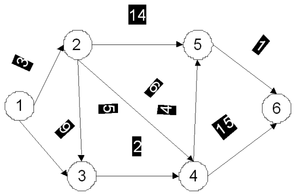


Figure 3. The knowledge map

For example, learner who has a threshold $\partial = 20$ for the required time attribute. With knowledge map is described in Figure 3. $V_s = \{1\}$, $V_e = \{6\}$, applying candidate learning path selecting algorithm, through six steps were described in Table 1.

We have the candidate learning path for learner is $1 \to 2 \to 4 \to 5 \to 6$ with required time has minimum value of 14. In case, the threshold ∂ is greater than minimum value which is the algorithm output result, the learner do not obtain his or her target in candidate learning path.

Table 1. Six steps of algorithm (for example)

Step	v	$V-S$	D	P
Init	1	$\{2,3,4,5,6\}$	$[3,9,\infty,\infty,\infty]$	[1,1,1,1,1]
1	2	$\{3,4,5,6\}$	$[3,8,9,17,\infty]$	[1,2,2,2,1]
2	3	$\{4,5,6\}$	$[3,8,9,17,\infty]$	[1,2,2,2,1]
3	4	$\{5,6\}$	[3,8,9,13,24]	[1,2,2,4,4]
4	5	$\{6\}$	[3,8,9,13,14]	[1,2,2,4,5]
5	6	-	[3,8,9,13,14]	[1,2,2,4,5]

4. BAYESIAN BELIEF NETWORK TO GENERATE LEARNING PATH

In this section, we describe the constructed Bayesian belief network which is based on some candidate learning paths. Our target is to generate a learning path that satisfies all of learner demands. Learning path is a set of vertices in knowledge map which are knowledge unit that learner need to browse.

4.1. Bayesian belief network

The underlying theory of BBN combining with Bayesian probability theory and the notion of conditional independence represents dependencies among variables. To date, BBN have proven useful in many areas of application such as medical expert systems, diagnosis of failures, pattern matching, speech recognition, and, more relevantly as risk assessment of complex systems in high-stakes environments. A BBN is a directed graph whose nodes represent the (discrete) uncertain variables of interest and whose edges are the causal or influential links between the variables. Associated with each node is a node probability table. This is a set of conditional probability values that model the uncertain relationship between the node and its parents together with any uncertainty that presents in that relationship [6, 7, 8].

4.2. Using Bayesian Belief Network to generate learning path

The learning path generating process includes two steps. The first step, we create node probability table Base on candidate learning path, we calculate probability value which denotes independently and represents in learning path. The second, we constructe a BBN to calculate probability value which represents for each knowledge unit in learning path.

4.2.1. Create knowledge unit probability table

Applying candidate learning path selecting algorithm mentioned in section 3, we have number of learning paths corresponding with LO's attributes which are assigned for adaptively. For example, with knowledge map in Figure 3, we have four candidate learning paths such as $1 \to 2 \to 4 \to 5 \to 6, 1 \to 3 \to 4 \to 5 \to 6, 1 \to 3 \to 4 \to 6$, and $1 \to 3 \to 2 \to 5 \to 6$.

Using this result, we create node probability table (NPT) for each knowledge unit. The values in node probability table show the knowledge unit probabilities which will present in learning path when the learner examines independent knowledge unit. In above example, we have NPT as follows:

Table 2. Probability for independent represent in learning path of each node

	1	2	3	4	5	6
True	1	0,5	0,75	0,75	0,75	1
False	0	0,5	0,25	0,25	0,25	0

In table 2, node 1 and 6 have probability value p set to 1 because these are starting node and target node of learning path. Node 2 represents in two candidate path, so p(2=true)=0,5, node 3 do not represent in one candidate path so p(3=false)= 0,25, and so on.

4.2.2. Constructing a Bayesian Belief Network

We construct knowledge map as a BBN with set of variables $X = \{X_1, X_2, ..., X_n\}$ consisting of a network structure S that encodes a set of conditional independence assertions about variables in X, and a set P of local probability distribution for X. The nodes in S are in one-to-one correspondence with the variables X. We use X_i to denote both variables and its corresponding node of knowledge map, and Pa_i to denote the parents of node X_i. In particular, given structure S, the joint probability for X is given by $p(x) = \Pi p(x_i|pa_i)$.

After construct BBN, we calculate the probability for one knowledge unit which represents in learning path. In above example,we do not calculate the probability of node 1 and node 6 because they are learner's starting point and target point so they are represented in learning path. All of probability value of remaining nodes in knowledge map will be calculated.

In Figure 3, node 3 depends on node 2, so $p(3|1, 2, 4, 5, 6) = p(3|2)$, and so on, we have $p(4|1, 2, 3, 5, 6) = p(4|2, 3)$, $p(5|1, 2, 3, 4, 6) = p(5|2, 4)$, $p(6|, 1, 2, 3, 4, 5) = p(6|4, 5)$.

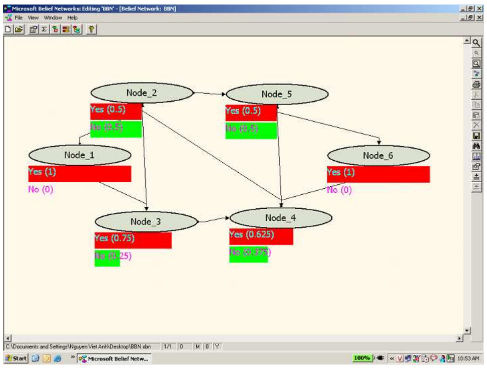


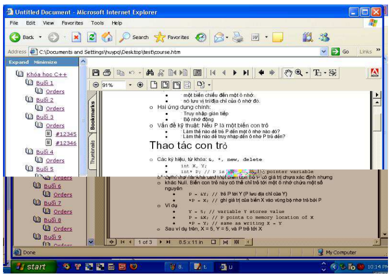
Figure 4. A bayesian belief network

In deployment, we use MSBNX tool to construct a BBN. Figure 4, knowledge map in example mentioned in section 3 is denoted. With this result, we easily has learning path is 1 → 3 → 4 → 6.

However, with deployed course, the BNN is quite complex, it includes more than thirty nodes, so we will improve our BBN and it is the topic to discus in the future.

5. IMPLEMENTATION

We built ACG system [1] based on web application. Each course has fifteen sections. Teachers or course designers sketch knowledge map for the course. When learners participate in the course, they was given a questionnaire and pre - test, this is a tool to obtain learner demands The ACG system use candidate learning path selection and BBN through MSBNX tool to generate the course that is appropriate for learners.



- một biến chiều đến một ô nhớ.
- nó lưu vị trí/địa chỉ của ô nhớ đó.

○ Hai ứng dụng chính:
- Truy nhập gián tiếp
- Bộ nhớ động

○ Vấn đề kỹ thuật: Nếu P là một biến con trỏ
- Làm thế nào để trỏ P đến một ô nhớ nào đó?
- Làm thế nào để truy nhập đến ô nhớ P trỏ đến?

Thao tác con trỏ

○ Các ký hiệu, từ khóa: &, *, new, delete
- `int X, Y;`
- `int* P; // P is` [illegible] `pointer variable`

○ [illegible] khai báo một biến con trỏ P có giá trị chưa xác định nhưng khác Null. Biến con trỏ này có thể chỉ trỏ tới một ô nhớ chứa một số nguyên
- `P = &Y; //` trỏ P tới Y (P lưu địa chỉ của Y)
- `*P = X; //` ghi giá trị của biến X vào vùng bộ nhớ trỏ bởi P

○ Ví dụ
- `Y = 5; // variable Y stores value`
- `P = &X; // P points to memory location of X`
- `*P = Y; // same as writing X = Y`

○ Sau ví dụ trên, X = 5, Y = 5, và P trỏ tới X

Figure 5. Interface of an adaptive course

Conclusion

In this paper, we describe learning path generating process for each learner based on learner's profile. To do this, we use the shortest path searching to create candidate learning paths for each supplement attributes of LO as well as learner's demands. With these paths, learners have many ways to browse the course by themselves. After that, we constructed a BBN to generate learning path. BBN based on knowledge map and node probability table which is the result of candidate learning path. This learning path is suggested for learner who will have the best way to participate in the course.

Acknowledgment. The work on this paper was partly supported by the research project No. 204006 entitled "Modern Methods for Building Intelligent Systems" granted by the National IT Fundamental Research Program of Vietnam.

REFERENCES

[1] P. Brusilovsky, Methods and techniques of adaptive hypermedia, *User Modeling and User Adapted Interaction* **6** (2–3) (1996) 87–129.

[2] Anh Nguyen Viet and Dam, H.S, ACGs: Adaptive Course Generation System - An efficient approach to build E-learning course, *Proceedings of the Sixth IEEE International Conference on Computers and Information Technology*, Seoul, Korea, 2006 (259–265).

[3] Anh Nguyen Viet, Dam, H.S, Applying weighted learning object to build adaptive course in E-learning, *Proceedings of the 14th International Conference on Computers in Education*, Bejing, China, 2006.

[4] Yanyan Li, Ronghuai Huang, Dynamic composition of curriculum for personalized E-learning, *Proceedings of the 14th International Conference on Computers in Education*, Beijing, China, 2006.

[5] C. Zhao, and L. Wan, A shortest learning path selection algorithm in E-learning, *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, Kerkrade, The Netherlands, 2006.

[6] V. Kumar, Rating learning object quality with distributed bayesian belief networks: the why and how, *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies*, Kaohsiung, Taiwan, 2005.

[7] Martin Neil, Norman Fenton, and Manesh Tailor, Using bayesian networks to model expected and unexpected operational losses, *Risk Analysis* **25** (4) (2005).

[8] David Heckerman, "A Tutorial on Learning With Bayesian Networks", Technical Report (1996) 95–06.

Nhận bài ngày 10 - 9 - 2007
Nhận lại sau sửa ngày 16 - 4 -2008