Adapting Learning Activities for Teaching Know-How in Adaptive Course Generation System

Viet Anh NGUYEN, Si Dam HO
College of Technology, Vietnam National University Hanoi, Vietnam
vietanh@vnu.edu.vn

Abstract: In our previous work, we have built an adaptive course generation system (ACGS) [1] to adapt a learning path for learning goals. In this paper we describe learning activity as represented in the learner model and domain model to teach learners how to get through course content based on learner’s knowledge. Therefore, our model aim is to develop Bayesian Network learner model to manage overlay knowledge model and adapt learning activities based on a task model. In addition, an implementation of this model with the course topic about computer science domain for third year students is also described.

Keywords: Learning activity, ACGS, Adaptive Hypermedia, Bayesian Network

Introduction

In this paper, we improve the ACGS model by describing learning activity represented in the learner model and domain model to teach learners how to get through course content based on learner’s knowledge. After they have been recommended the learning path, the system will help them how to acquire each concept in the learning path. Having different knowledge level, each learner can achieve various concepts to get through learning goals. Therefore, the system will point out activities which learners need to get in order to know how to understand knowledge of concepts as well as to know how to finish learning activities. Bayesian network will evaluate activity based on variable states and learner model to decide which activity learners need to do or not. Because of activity’s relationships, the model also gives step-by-step adapted instructions for each learner.

1. Learning activities task model for teaching know-how

1.1 Course domain model

1.1.1 Concepts

The course domain is represented as a direct acyclic graph (DAG) with several nodes and vertex connecting between them. Node depicts an atomic concept while vertex depicts prerequisite relationship among the concepts. Clearly, in order to acquire a topic, those concepts are needed to acquire. In this model, we defined some terms as followings:
- Concept is an atomic unit that represents the knowledge content unit of the course.
- **Prerequisite concept**: Concept $C_i$ is a prerequisite concept of concept $C_j$ only if the learner acquire concept $C_i$ is the concept $C_j$ acquired.

1.1.2 **Learning activities task**

A task statement refers to a set of coherent activities that are performed to achieve a goal in a given domain. The mechanism of hierarchical and recursive decomposition of a problem into sub-problems is one of the basic characteristics of the hierarchical task model [5].

In our model, we define *Learning Activity Task is a learning activity which learner need to do in order to acquire a concept or finish learning goal.*

1.1.3 **Bayesian Network learner model**

We used an overlay model to describe learner models, in which each concept and learning activity in domain model has a value to measure learner’s knowledge about it. Instead of using vector or concrete numeric value, we use probability value to evaluate a learner’s knowledge. We assign a set of variable to measure it with three states: not acquired, in progress, acquired. $p(\text{not-acquired}(C))$ represents probability value of not acquired state for concept $C$, $p(\text{in-progress}(C))$ denotes probability value of in progress state for concept $C$, and $p(\text{acquired}(C))$ denotes probability value of acquired state for concept $C$. There is $p(\text{not-acquired}(C)) + p(\text{in-progress}(C)) + p(\text{acquired}(C)) = 1$.

In our model, we only take into account modeling prerequisite relationship between concepts which is important both for instructional planning purposes and for gathering information about the current state of learner knowledge. We focus on evidence of learner knowledge relationships. Suppose that $A$ is prerequisite of $B$, when $A$ is acquired, $p(\text{acquired}(B)|\text{acquired}(A))$ probability values will estimate learners knowledge about concept $B$ based on concept $A$.

Through questionnaire and assignment, our model has gained evidence value about learner’s knowledge for each concept. Based on NPT, BN can evaluate learner’s knowledge for all concepts in domain model because of their relationships. Each of the activity tasks has three states: not-finished, in-progress, and finished. The results of what the learner has done to be a factor to recognize if the task is finished or not.

Similar to variable of measuring learners knowledge, if $T_1, T_2, \ldots, T_n$ is activity tasks for the abstract task $T$. The value to estimate whether the task $T$ is finished or not is computed by: $p(\text{finished}(T)|\text{finished}(T_1), \ldots, \text{finished}(T_n))$ with symbol , denotes AND operation.

2. **Adaptation Engine**

Adaptation is the process to select activity tasks for each learner based on learners’ model. The learner with different knowledge levels need to do different tasks including abstract and activity tasks to finish learning goals. The adaptation process selects learning resources through several phases as denoted in Figure 2.

![Figure 2: Adaptation engine](image-url)
After selecting the best learning path for each learner, the system will select learning activities that the learner should finish to acquire concepts. Indeed, for each nodes of the learning path, system will point out how to acquire concepts by giving learning activity that should be done by the learner. What activity that learner needs to do is based on evaluation of learners’ knowledge and decision of probability values that measure learner knowledge about learning tasks. Task T need to be done or can be omitted is decided by probability values that measure learner knowledge about tasks T₁, T₂, ..., Tₖ is prerequisite tasks for T. This value is computed by three formulas: 

\[ p(\text{finish}(T)|\text{finish}(T₁), \text{finish}(T₂), \ldots, \text{finish}(Tₖ)) \]

for finish state,

\[ p(\neg\text{finish}(T)|\neg\text{finish}(T₁), \neg\text{finish}(T₂), \ldots, \neg\text{finish}(Tₖ)) \]

for not-finish state and

\[ 1 - p(\text{finish}(T)) - p(\neg\text{finish}(T)) \]

for in-progress state. If probability value for finished state is greater than \( α \) threshold, the learner can omit the task T.

3. Our Experiment

We designed a course “How do you design a relationship database?” for third year students. To design a database, first of all, a student need to skim problem specification and then participate in four phrases: designing entity relationship diagram, transforming entities relationship diagram to physical tables, normalizing tables, and defining queries to retrieve information. There are twenty six concept nodes in the course model. Furthermore, a task diagram is composed of twelve abstract tasks and twenty-nine activity tasks. There are two kinds of activity tasks: consequent and parallel. The former requires carrying out in order; the later can be done at the same time.

4. Conclusion

Main contribution of this paper is learning activities model based on Bayesian network to select adaptive learning activities for each learner. Evaluating learner’s knowledge as well as variable learning activity’s state, our system selects essential activities for learners. As a result, the learner not only has recommended concepts to learn but also have recommended learning activities to do. Result of experiments pointed out selected adaptive learning activities which meet learner’s knowledge and support them step-by-step to acquire concepts.

Acknowledgements

This work is partly supported by the research project No.QC.09.09 granted by Vietnam National University, Hanoi and This work is partly supported by a research grant of TRIG project at University of Engineering and Technology, Vietnam National University, Hanoi.

Selected References
