Analysis of Productive Learning Behaviors in a Structured Inquiry Cycle Using Hidden Markov Models

Hogyeong Jeong, Gautam Biswas, Julie Johnson, and Larry Howard
{hogyeong.jeong, gautam.biswas, julie.l.johnson, larry.howard}@vanderbilt.edu
Institute for Software Integrated Systems, Vanderbilt University

Abstract. This paper demonstrates the generality of the hidden Markov model approach for exploratory sequence analysis by applying the methodology to study students' learning behaviors in a new domain, i.e., an asynchronous, online environment that promotes an explicit inquiry cycle while permitting a great deal of learner control. Our analysis demonstrates that the high-performing students have more linear learning behaviors, and that their behaviors remain consistent across different study modules. We also compare our approach to a process mining approach, and suggest how they may complement one another.

1 Introduction

One of the broader goals for Intelligent Learning Environments (ILEs) is to help students not only become proficient in particular topics, but also to become better prepared for future learning (PFL) [3]. Among others, two PFL considerations stand out: (1) students should learn the content with good understanding so that they may apply it in problem solving situations, and (2) students should develop good learning behaviors and strategies that they can apply in other learning situations, even those outside the computer learning environment. Computer tutoring systems typically impose learning or problem solving structure on the learner [1]. They are primarily designed to monitor students' performance on the task, and provide feedback to help students improve their learning performance. In contrast, students have choice in their learning activities in inquiry-based and exploratory learning environments. In such systems, it may be quite useful not only to monitor students' learning performance, but also track and interpret their activities, map them into good and bad learning behaviors, and then provide appropriate feedback to help the students become better learners. If done well, this would also help students become better prepared for future learning. In this paper, we employ data mining methods to analyze students' activity sequences and map them on to potential learning behaviors.

We have used hidden Markov models (HMMs) [6, 7, 10] as an exploratory tool to analyze traces of students' activities as they learn by teaching a computer agent called Betty. Exploratory data analysis makes few initial assumptions about the domain, but its results, even if they are preliminary, informs confirmatory data analysis. The system, Betty's Brain [2], is also designed to help students become aware of and learn self-regulated learning (SRL) skills through interactions with the teachable agent, Betty, and a mentor agent, Mr. Davis. The HMMs provide aggregated descriptions of the more prevalent patterns of students learning behaviors (both good and bad). A challenging task, after learning a HMM from the trace data, is the interpretation of its "hidden" states. In our work, we have interpreted the states in terms of the set of activities linked to the

states, and then characterized activities associated with states as learning behaviors. However, how best to assign meaning to the states in HMM analyses is application dependent.

In this paper, we demonstrate the generality of our HMM approach by applying it to analyze student learning behaviors in another domain, an asynchronous cyber learning environment in cyber terrorism. Unlike the middle school subjects in the Betty's Brain studies, the users of this system are adult professionals, who use this course as a degree requirement or for professional certification. A few use the system for self-study. The system had a wide variety of users (over 6,000) but we do not have access to the specifics of their background information. Therefore, we decided to focus our exploratory data analysis methods on examining the differences in behaviors between the high and the low performers on the system, where the categorization of high versus low was determined from the posttest scores. We believe that this analysis will provide insights on how learning behaviors relate to performance. As further discussion, we will compare our HMM analyses with the results of a model-driven analyses conducted by our colleagues at ISIS who developed this system [5]. In particular, we note how the process modeling techniques can complement our analysis by providing a finer-grained perspective on students' behaviors.

2   System Description

The asynchronous cyber learning environment on cyber-terrorism is derived from the STAR Legacy Cycle, a software shell designed to organize learning activities as an inquiry cycle [9]. A key feature of this learning model is that it makes the steps in the learning cycle explicit, making it easier for learners to understand the intent of the activities, and how they apply to learning. Some adaptations were made to the original STAR Legacy cycle to tailor it for asynchronous, online learning of a number of cyber-terrorism modules. The adaptation preserves the essential qualities of the explicit inquiry cycle: adaptive and progressive feedback, and scaffolds for planning, reflection, and synthesis. The *Overview* (O) phase is the portal that students use to enter and exit the cycle. The modified cycle is represented in five phases (see Fig. 1): (1) *Challenge* (C), where the students are presented with the problem description; (2) *Initial Thoughts* (T), where the students provide their initial thoughts on the problem; (3) *Resources* (R), where the students can learn about the problem; (4) *Self-Assessment* (A), where the students answer assessment questions to get formative feedback on their progress; and finally (5) *Wrap-up* (W), where the students can review their initial thoughts and conclude the module.
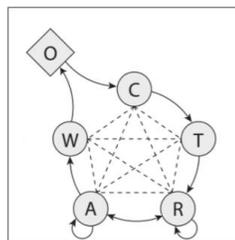

Figure 1. The Adapted STAR Legacy Cycle

While the canonical inquiry cycle supported by the STAR Legacy interface has students progressing in a linear fashion from *Challenge* to *Wrap-up*, the system is also designed to promote the idea of learner control [4, 11]. In other words, students are given complete freedom to navigate through the cycle to accomplish their learning tasks. We should note that it is precisely this freedom that allows for various behavior patterns to emerge as the students use the system; this makes the activity sequences collected from the system particularly amenable to our exploratory data analysis methods.

3 Experimental Study

The users of the system are primarily adults who take the course to meet certification requirements or for recommended workplace training. A few students took the course for other reasons, but this number was insignificant. In all, the preliminary dataset which we retrieved consists of 6,298 learners covering 50 learning modules in 10 independent-study courses.

Each module represents roughly 4 to 8 hours of self-study, and each independent-study course contains 3 to 7 different learning modules. The students could pass the pretest for a module and avoid the training. Otherwise they went through the training and took the summative assessment at the end (outside of the training system). Both the pretest and the summative assessment questions were randomly generated subset from a shared pool of questions. It is important to note that learners who successfully completed every section (pretest or summative assessment) were immediately awarded the course certification. In fact, around 18% of the students successfully prequalified during the pretest, and skipped the training modules. Hence, the students who used the system were necessarily those who did not have the requisite prior knowledge to pass the test the first time.

In our previous analyses, we created groups based on how the students improved from pre to posttest by using the system, and made distinction between a low (pretest performance) and low (posttest performance) group, low and high group, and high and high group [2]. However, in this study, as discussed above, all students who use the system universally belong in the "low prior knowledge" category. Therefore, our grouping uses only the posttest scores. For this study, we split the scores into three uniform groups (after determining that the data was unimodal) to define our low, middle and high performance groups. At the two ends of the middle group, there were a number of scores that were close to the low and high group scores respectively, so we ignore the middle group, and focus our analysis on the differences between the low and the high groups.

Using these performance groupings, we examine ways to characterize the behaviors of the different students, and determine, for example, whether the high-performing students have learning behaviors that are distinct from the low performers. In our previous domain, we have found that the higher-performing students exhibit patterns of behaviors that correspond to good learning strategies (e.g., explore one's understanding of a topic by asking relevant follow-up queries and checking to see if the explanations for the answer are meaningful) [7]. For our current domain, cyber security, we direct our attention to studying how students transition between the different phases of the STAR

Legacy cycle. In particular, we are interested in determining whether the path students take when following the cycle is in any way related to the students' performance. For example, one may argue that the students who follow the cycle in a linear fashion (the implicit model of inquiry learning in STAR Legacy) would perform better in their learning tasks than students who make numerous jumps between the phases. An alternate hypothesis may state that students who make multiple forward and backward jumps in the cycle do so because they have formulated their own strategy for learning the content material, and hence would perform better.

Our behavior analysis will focus on the transitions students make as they go through the cycle and their context. Howard, et al. [9] have defined six different types of transitions: *linear* (L), *jumping* (J), *retrying* (R), *searching* (S), *transitioning* (T), and *backtracking* (B). *Linear* transitions comprise directed navigation steps in the cycle, and may be performed by clicking the "Next Up" navigation button appearing at the end of each activity or resource. *Jumping* transitions imply the use of Resources and Assessments in a non-linear way through the use of descriptive menus. *Retrying* transitions occur in the Assessment phase, and represent the activity of re-attempting a specific question. *Searching* transitions comprise of actions, where the students search the materials in either the Resources or the Assessments phase. *Transitioning* represents the students moving to and from the course module menu. Finally, *Backtracking* transitions indicate that the student has previously seen the destination activity, resource, or menu.

We hypothesize that students in the different performance groups will employ different learning behavior patterns (and likely, different strategies) when using the system. Further, students' behavior patterns will evolve as they study different modules. Therefore, hypothesis 1 states that there will be a marked difference between the two performance groups, especially in their use of linear transition behaviors as they learn using the STAR Legacy cycle. Hypothesis 2 states that the students' behaviors will evolve as they use the system, and that this can be seen when comparing early and late module behaviors. We will use two metrics to assess the difference between the models: (1) the definition and interpretation of the HMM states; and (2) the stationary probabilities, which represent the proportion of time spent in a state relative to the other states of the HMM. We will describe these metrics in more detail in the following section.

4   Methods

Before the students' trace data can be analyzed to generate HMMs, necessary preprocessing steps have to be executed on the raw activity sequences. As stated before, the focus of our analysis is to interpret students' behaviors as states and the transitions between states. Thus, the objective of the preprocessing is to extract and reduce the activity sequences to just the transition and the context associated with the transition. Because clear distinctions were made between the different types of transitions, all of the preprocessing can be done automatically. At the end of the preprocessing phase, we have activity sequences consisting of series of transition-context pairs (e.g. "AR-Linear; RR-Linear; RR-Retrying", where A represents the *Assessment* phase, and R represents the *Resources* phase).

4.1 The HMM Procedure

Deriving a HMM from students' activity sequences should result in a model that identifies frequently occurring sequence patterns, which are then interpreted as student behaviors. The HMMs are so named because the derived states are hidden, i.e., they cannot be directly observed. Instead, one provides meaning or interpretation to the states by studying the activities associated with the state. These activities imply behaviors, and the collection of states may imply one or more behavior patterns.

A first step in model generation is to initialize the parameters that define the states of the HMM and the possible state transitions. Starting from an initial model description, expectation-maximization techniques are applied iteratively until the model parameters converge [6, 10]. The estimation process is quite sensitive to the initial state description, and bad initializations may lead to generating suboptimal models. In past work [6, 8], we have used a conceptual clustering algorithm to generate good parameters for the initial state.

Another unknown in the model derivation process is the "best" number of states that define the HMM. The metric that we use to assess the model fit is the Bayesian Information Criterion (BIC) measure, which takes into account both the log likelihood of the model (i.e., how likely the model is given the data) and the number of states in the derived model (i.e., how complex the model is) to find the model that strikes the best balance between high likelihood and low complexity [8]. The full procedure for deriving HMMs from sequence data is outlined in our previous EDM paper [6].

The HMMs offer us a state-based aggregated interpretation of the students' activity sequences, and we can analyze them in various ways. A first step is to assign meanings or labels to the states of the model. For example, we may find that some types of activities tend to cohere in certain states, which would provide clues to determine what behaviors these states represent. Also, we can examine the stationary probabilities to get a sense of what the more relevant (i.e., more frequently occurring) states are. The stationary probability, as we define here, is the relative proportion of activities that belongs to a certain state (i.e., the stationary probability of a state A is the proportion of occurrences of State A among all states that occur in a sequence of length n iterations generated by the model; n is typically the average number of activities in the input sequences). For example, a state having a 20% stationary probability implies that 20% of students' activities during the session are related to the behavior(s) that state represents. We can also study the transitions between the states to see how students transition between different behaviors. In addition, the structures of the models themselves can be studied to see if certain patterns appear (e.g., cycles), and these patterns can then be interpreted in terms of students' learning behaviors.

4.2 The Experimental Datasets

The HMMs were run using two different datasets. For the first dataset, we investigated the first module that the students worked on. The second dataset was a module that the

students worked on toward the end of their study for that course. Table 1 lists the number of students in the early and late modules, and the average posttest scores for these modules.

Table 1. Modules Used

	Module Type	Numbers (low, high)	Average Scores (low, high)
First Dataset	Early	(189,83)	(51, 95)
Second Dataset	Late	(133,97)	(34, 93)

5   Results

Examples of the HMM structures generated are shown in Figs. 2 and 3. All four models (Low-Early; Late-Early; High-Early; and High-Late) had 5 or 6 states. The main difference among the models was more in the activities associated with each state, rather than the state transition behavior. We discuss one of the derived models in some detail to familiarize the reader with the HMM structure. Fig. 2 depicts the model for the high performance group in the early module. Table 2 complements the model by listing the major composition of activities for each state. The first two letters denote the phases involved in the transition (e.g. "AR" is the transition from *Assessments* to *Resources*), and the letter after the hyphen indicates the transition type (e.g. "L" is linear).

We see that the students begin from state 1, and moving from Overview to Thoughts is the only activity associated with this state. Fig. 1 confirms that this corresponds to the initial part of the STAR Legacy cycle, so we label this state as the *Start State*. From state 1, the students move directly to state 2, where they proceed to thoughts (82% of the time) or jump to the resources (14% of the time) (see Table 2). Since this corresponds to the start of the problem solving, we call this the *Initiation State*. Then, the students move to state 3, where the students perform multiple activities, but the two primary ones are proceeding linearly through the resources (56%) and retrying questions in the resources (13%). The dominant activities plus the others lead us to label this state as the *Assessment State*.

Other analyses have determined that students spend significant portion of the time working on assessments (can be thought of as self-assessments), so not surprisingly our model indicates a self-loop with high likelihood (90%) for this state. When students exit the *Assessment State*, the HMM indicates moves to state 4 (likelihood of 7%) or to state 5 (likelihood of 3%). State 4 is composed mainly of linear transitions involving the wrap-up phase (47%) and retrying assessments (35%). For this reason, we call this the *Wrap-up State*. States 5 and 6 are composed mainly of backtracking transitions. In particular, we note that state 5 contains a significant portion of backtracking through resources and assessments (44%). Hence, we call state 5 a *RA (Resources-Assessment) Backtracking State*. Meanwhile, state 6 is composed significantly of activities involving wrap-up, overview, or thoughts (48%). Hence, we call state 6 a *HL (High-Level) Backtracking State*.
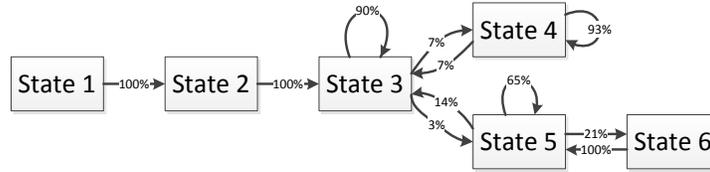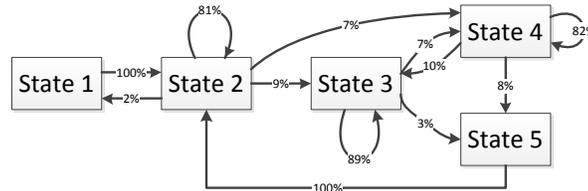
Figure 2. Early Module – High Performance Group

Figure 3.Early Module – Low Performance Group

Table 2. Distribution of Major Activities (those with probabilities greater than 5%)

	State 1	State 2	State 3	State 4	State 5	State 6
Early-High	OC-L (100%)	CT-L (82%)	AA-L (56%)	AA-R (35%)	RA-B (24%)	WO-B (26%)
		CR-J (14%)	AA-R (13%)	WO-L (25%)	AR-B (20%)	RA-B (13%)
			RA-J (8%)	AW-L (22%)	RR-B (8%)	CT-B (12%)
			TR-L (6%)	AR-J (6%)	WO-L (6%)	AA-B (11%)
			RR-L (6%)			WA-B (10%)
Early-Low	OC-L (100%)	RR-L (19%)	AA-L (45%)	AR-B (32%)	AW-L (51%)	
		CT-L (11%)	AA-R (35%)	RA-B (31%)	AW-B (19%)	
		WO-L (10%)		AA-B (14%)	AW-J (12%)	
		TR-L (10%)		AA-L (8%)		
		WO-B (8%)		RA-J (6%)		

Figure 3 shows the corresponding HMM for the low performing group working on the early module. This model has 5 states. Upon close observation, we see that the reduction in states is due to the *Wrap-up State* and the *HL Backtracking State* being merged into a single state (State 3). The other states that we have identified all appear in this model, albeit with a somewhat different compositions of activities, which we will be one of the key issues in our investigation. The identified states are summarized in Table 3.

Using these interpreted states, our first level of analysis would be to see if there are differences in the proportion of time that the students spend in each of the states. The computed stationary probabilities are summarized in Table 4. It is clear that the High-Early and the High-Late models are almost identical. Thus we may conclude that the high-performing students' learning behaviors remained the same as they studied the different modules. On the other hand, the low performers' behaviors changed significantly from the early to the late module, primarily in the amount of time they spent on wrap up, as well as the amount of backtracking between resources and assessments.

The mixed state that was observed early also went away in their learning activities in the late module. One may conclude that with time, the low performers started behaving more like the high performers, and one may conjecture that this was because the system was helping the low performers become better learners. This needs to be investigated further. In terms of hypothesis 2, Table 4 implies that the low-performer data supports hypothesis 2, but that the high-performer data does not. One may conclude that this is reasonable, because the high performers have figured out how to do well, and do not need to modify their behaviors.

Table 3. Descriptions of States

Start State	Students move from the *Overview* to the *Challenge* phase
Initiation State	Students move from the *Challenge* to mainly the *Initial Thoughts* phase or the *Resources* phase
Assessment State	Students stay in the *Assessments* phase, mainly moving to the next question, or retrying a question they get wrong
Wrap-up State	Students proceed to the *Wrap-up* phase and conclude the module
RA (Resources-Assessment) *Backtracking* State	Students revisit previous resources or assessment questions
HL (High-Level) *Backtracking* State	Students revisit *Initial Thoughts* or the *Challenge* phases

Table 4. Stationary Probabilities

	Start	Initiation	Assessment	Wrap-up	RA Backtrack	HL Backtrack	Merged Wrap-up and RA Backtrack
Low-Early	3	-	40	2	29	-	25
High-Early	2	2	50	33	10	2	-
Low-Late	2	2	37	12	46	-	-
High-Late	2	2	50	33	11	-	-

Another interesting difference revealed by the models is how the proportions of activities significantly differ among the different performance groups. In particular, there is much higher incidence of linear transitions in the *Initiation* State and the *Assessment* State among the higher performing groups. These results are summarized in Table 5.

6 Discussion and Comparisons

Using the results from the hidden Markov model analysis, we sought to investigate how the high-performing students transitioned through the different phases in the system in contrast with the low-performing students. In particular, we were interested in the students' relative tendencies to follow the explicit, canonical model presented by the system, given the great degree of freedom that they were also provided. Confirming our first hypothesis, we found that the high-performing students moved more linearly from

the *Challenge* to the *Initial Thoughts* phase, and also that they moved more linearly through the *Assessment* phase. We also found that the high-performing students spent markedly more time in the *Wrap-up*state, while the low-performing students engaged much more in backtracking through previous assessment questions and resource items. While these results are promising, we should note that the results likely stem from the high-performing students having a better understanding of the domain, and, therefore, showing better abilities to plan their learning and assessment tasks. On the other hand, the low performers seem to flounder more, which manifest as backtracking actions.

Table 5. Proportions of Linear Transitions

	Low-Early	**Low-Late**	**High-Early**	**High-Late**
Initiation State	-	75	82	82
Assessment State	45	52	68	68

Our results are complemented by our colleagues' findings from using process mining techniques to examine similar problems [5]. This approach uses specifically structured models to investigate individual problems at a greater detail. For example, our colleagues provide a Petri Net Model for how the learners navigate through the STAR Legacy cycle. Such a model provides precise numbers of students who follow the sequential path and those who deviate from it at any point. A result of interest to us is the finding that most students follow the canonical model at first, but start to diverge as they use the system. Of particular concern is that most of the divergence stems from the students' avoidance of the *Thoughts* phase in the later modules; the *Thoughts* phase figures prominently into the cycle as it is designed to create "cognitive dissonance" among the learners [4]. Complemented with our finding that higher-performing students are more likely to proceed linearly and not skip the *Thoughts* phase. It might be worth investigating why the students grow to avoid the *Thoughts* phase, and whether this avoidance actually leads to a drop in performance. Also, the direction of the causality should be explored: does proceeding linearly lead to higher performance, or does higher cognitive ability lead to proceeding linearly? Our hypothesis is that learners face increased cognitive load for the metacognitive tasks, such as strategic planning, and that following the structural scaffolds in the environment, i.e. progressing linearly, would help them manage the cognitive load, and hence perform better.

7 Conclusions and Future Work

In this paper, we have shown that the hidden Markov model approach is general, and can be applied to performing exploratory data analysis in new domains. The analysis implies that there are differences in transitionary state behavior between the high and the low-performing students. Namely, the results indicate that the high-performing students move through the model more linearly and spend less time backtracking than the low-performing students. Our results, combined with our previous work, show the general nature of the hidden Markov model approach for exploratory sequence analysis. Like speech synthesis applications [8], HMMs may prove to be a valuable tool for analyzing learning behaviors in a variety of domains. We have also shown how exploratory data

analysis can be complemented by process mining analysis to help examine problems in greater detail. In particular, we imagine that interesting and inconclusive results from exploratory data analysis may be investigated using process mining to yield more definite findings. We believe that the combination of these approaches will lead to many promising avenues to extend our research in the future.

References

[1] Biswas, G., Leelawong, K., Schwartz, D., & Vye, N. (2005). Learning by Teaching: A New Agent Paradigm for Educational Software. *Applied Artificial Intelligence, 19*, 363-392.

[2] Biswas, G., Roscoe, R., Jeong, H., & Sulcer, B. (2009). Promoting Self-Regulated Learning Skills in Agent-based Learning Environments. Hong Kong: Proceedings of the 17th International Conference on Computers in Education.

[3] Bransford, J., & Schawrtz, D. L. (1999). Rethinking transfer: A simple proposal with multiple implications. *Review of Research in Education, 24*, 61-101.

[4] Chen, M. (1995). A methodology for characterizing computer-based learning environments. *Instructional Science*, 183-220.

[5] Howard, L., Johnson, J., & Neitzel, C. (2010). Examining learner control in a structured inquiry cycle using process mining. *3rd Int. Conf. on Educational Data Mining.* Pittsburgh.

[6] Jeong, H., & Biswas, G. (2008). Mining Student Behavior Models in Learning-by-Teaching Environments. *Educational Data Mining*, (pp. 127-136). Montreal.

[7] Jeong, H., Gupta, A., Roscoe, R., Wagster, J., Biswas, G., & Schwartz, D. (2008). Using Hidden Markov Models to Characterize Student Behavior Patterns in Computer-based Learning-by-Teaching Environments. *Intelligent Tutoring Systems*, (pp. 614-625). Montreal.

[8] Li, C., & Biswas, G. (2002). A Bayesian Approach for Learning Hidden Markov Models from Data. *Special issue on Markov Chain and Hidden Markov Models, Scientific Programming, 10,* 201-219.

[9] Neitzel, C., Johnson, J., & Howard, L. (2010). Behavioral data and indications of self-regulation: What learner choices reveal about their strategies. *Advances in Engineering Education*.

[10] Rabiner, L. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. IEEE, 77*(2), 257-285.

[11] Schwartz, D., Lin, X., Brophy, S., & Bransford, J. (1999). Toward the development of flexibly adaptive instructional designs. In C. Reigeluth, *Instructional-design Theories and models: A New Paradigm of Instructional Theory II* (pp. 183-214).