Comparative Action Sequence Analysis with Hidden Markov Models and Sequence Mining

JOHN S. KINNEBREW and GAUTAM BISWAS, Vanderbilt University

Computer-based learning environments produce a wealth of data on student learning interactions. This paper presents an exploratory data mining methodology for assessing and comparing students' learning behaviors from these interaction traces. In the first phase of this methodology, hidden Markov models (HMMs) are generated to model learning behaviors of the student groups being compared (e.g., high versus low performers). In this paper, we supplement the HMM technique with a novel combination of sequence mining techniques to identify differentially frequent patterns (between the student groups) in a finer-grained analysis. We demonstrate the complete methodology through the analysis of learning trace data from a recent middle school classroom study with the Betty's Brain learning environment. The results illustrate the effectiveness of this exploratory methodology and suggest further refinements of the HMM generation and sequence mining techniques.

1. INTRODUCTION

Computer-based learning environments (CBLEs) allow researchers to track many details of students' learning interactions and activities. This wealth of data provides an opportunity to more accurately assess, model, and understand student learning behaviors and strategies. The learning activities logged by a CBLE result from a variety of internal cognitive/metacognitive states, strategies, and processes used by the student. Employing a direct representation of these internal states and strategies with a probabilistic automaton, such as a hidden Markov model (HMM) [Rabiner 1989], has the potential for facilitating identification, interpretation, and comparison of student learning behaviors. Like a student's mental processes, the states of an HMM are hidden, meaning that they cannot be directly observed, but they produce observable emissions (*e.g.*, actions in a learning environment).

In this paper, we extend a modeling technique using HMMs for identification of group learning behaviors with a second phase employing a novel combination of sequence mining techniques to identify differentially frequent patterns between student groups. While HMMs provide an effective method for identifying overall learning behaviors in a group of students, less common behaviors and specific variations in ordering actions can be overlooked at this level of analysis/modeling. Our comparative sequence mining technique provides a finer-grained analysis of learning behavior *differences* between groups of students that complements the HMM techniques. The integrated methodology is illustrated by analyzing learning activity data from a recent middle school class study with the Betty's Brain learning environment. These results show the effectiveness of this exploratory methodology in a comparative analysis of learning behaviors.

2. DATA MINING TO ASSESS LEARNING BEHAVIORS

In recent years, a number of researchers have had success in applying HMM generation techniques to learning interaction data. For example, Pardos and Heffernan [2011] have successfully applied HMMs (in conjunction with a bagged decision tree classifier) to predict correctness of student answers in computer math learning environments. In contrast to research using HMMs for action prediction, our exploratory methodology uses

This work has been supported by NSF-IIS Award #0904387.
Authors' address: Department of EECS and ISIS, Vanderbilt University, 1025 16th Ave S, Ste 102, Nashville, TN 37212; Email: john.s.kinnebrew@vanderbilt.edu

HMM generation to identify learning behaviors of student groups and to compare across those groups. Boyer et al. [2009] have used HMMs for identifying strategies and high-level flow in student dialogues with human tutors. Cohen and Beal [2009] have used HMMs to analyze dialogues with an intelligent tutoring system to identify different states of engagement.

Some of the more closely related work with HMMs for identification of student learning strategies is the HMM clustering technique employed in [Shih et al. 2010]. Performance and learning metrics are used to direct generation of HMM collections that predict learning outcomes. In contrast, Biswas et al. [2010] generate HMMs from a set of student activity sequences to build a model that represents an "averaged" view of the group's learning behaviors. Previous studies have focused on determining how learning environment and scaffolding design assist students in learning science. In this paper, the HMM methodology is employed to identify and distinguish learning behaviors exhibited by high- and low-performing students.

The HMM generation techniques for identifying or comparing student learning behaviors, states, and strategies, use a relatively course-grained interpretation. An "averaged" group model allows for more effective interpretation of common learning behaviors and strategies. This level of modeling is important for providing an overview of learning behaviors in student groups and for efficiently tracking student behavior online to facilitate directed feedback.

To identify additional details of students' learning behaviors, we employ a combination of sequence mining techniques. An important part of this extended analysis is the use of sequential pattern mining [Agrawal and Srikant 1995] to identify frequent[1] patterns of actions within a group. However, this can also result in a very large number of frequent patterns, especially when allowing gaps to account for "noise." For example, over 1,000 patterns were found to occur in at least 80% of the 22 student interaction traces from the Hi performance group discussed in Section 3, even when limiting gaps to a single action between consecutive actions in the pattern. Employing a higher frequency threshold (*e.g.*, considering patterns exhibited by at least 95% of the students) can reduce this number but can easily overlook less common patterns that may imply important learning strategies in the group. In general, a major challenge in effective analysis of frequent patterns is limiting the often large set of results to interesting or important patterns (*i.e.*, the "effectiveness" of a mining technique as opposed to its "efficiency" in finding frequent patterns [Agrawal et al. 1993]).

To find these important patterns in a comparative analysis between student groups, our methodology combines sequential pattern mining with episode discovery/mining to identify *differentially frequent* patterns. In contrast to sequential pattern mining, episode mining defines frequency as the number of times the pattern can be matched *within* a single sequence of actions [Mannila et al. 1997]. Although generally considered distinct subfields of sequence mining, there has been some recent work in combining techniques and concepts from sequential pattern mining and episode mining. Like our approach, Ding et al. [2009] employ episode mining over a set of sequences, rather than a single sequence. However, our methodology separately employs sequential pattern mining to first identify common patterns within each comparison group. Lo et al. [2008] employ multiple frequency thresholds and techniques (from sequential pattern mining and episode mining) in a combined algorithm for the discovery of recurrent rules in a single set of sequences.

3. COMBINED HMM AND SEQUENCE MINING METHODOLOGY

Our exploratory data mining methodology combines data pre-processing, HMM generation, and sequence mining techniques for comparative analysis of learning behaviors across student groups. In this section, we describe the three major components of this methodology. To illustrate this methodology, we present a case study employing these techniques on data from a recent study in a middle school science classroom with the Betty's Brain learning environment.

[1]Pattern frequency in sequential pattern mining is defined as the number of different sequences (in this case the number of different student interaction traces) that include the pattern.

3.1 The Betty's Brain Learning Environment

The Betty's Brain system implements the learning-by-teaching paradigm to help middle school students develop cognitive and metacognitive skills in science and mathematics domains [Biswas et al. 2005]. Students interact with a Teachable agent (Betty) and a Mentor agent (Mr. Davis) to learn and understand a science topic. The system includes a set of indexed, hypermedia resources that students can access and use at any time while working on the system. Using the information from the resources, students explicitly teach the Betty agent by constructing a causal concept map representation [Leelawong and Biswas 2008]. Students teach Betty about new concepts and relationships (links) using a visual interface that includes menu selections and templates for adding and modifying information (*e.g.*, the interface includes the buttons: Teach Concept, Teach Link, Delete, and Edit).

Since our middle school students are novices in both the science topics and the teaching tasks, we provide them with a variety of scaffolds to help them overcome obstacles they may face in learning and teaching the domain material. In addition to answering queries, providing explanations, and taking/administering quizzes, the agents also provide spontaneous feedback to the student on the relative effectiveness of their teaching performance and applicable strategies. This feedback is designed to help students develop and employ self-regulated learning (SRL) strategies and metacognitive skills [Wagster et al. 2007].

To illustrate our methodology, we use interaction trace data from a recent study with 32 8th-grade students in a middle Tennessee science classroom. The average length of the student interaction traces in this study was 293 actions with a standard deviation of 95.2 actions. At the beginning of the study, students were introduced to the science topic (global climate change) during regular classroom instruction, provided an overview of causal relations and concept maps, and provided hands-on training with the system. For the next five days, students taught their agent about climate change and received feedback from both agents. In this version of the system, the majority of the SRL and metacognitive feedback was for knowledge construction strategies [Biswas et al. 2010]. The Mentor agent also provided some advice on monitoring strategies to help students recognize and correct errors in their concept maps.

The results of this study presented an interesting dichotomy in student performance at constructing their causal concept maps. 22 of the students finished the study with a correct map (*i.e.*, the map matched the expert map and their Teachable agent got a perfect score on the quiz). The other 10 students had an average map score[2] of 8.2 (out of 18), and the highest map score was 13. To illustrate our exploratory methodology, we divide the interaction traces of these students into two groups: (1) the "Hi" performers with correct final maps, and (2) the "Lo" performers with incomplete and incorrect final maps.

Before analyzing the sequences of student actions, we consider the evolution of their causal map structures to better understand how their actions might have contributed to their performance. Figure 1 provides representative examples of map evolution in the Hi (left graph) and Lo (right graph) groups. The x-axis corresponds to the cumulative number of map edits, and the beginning of distinct sessions are marked by the date. In general, students in both groups had a monotonically increasing number of correct links in their maps, with minor downward fluctuations that were quickly reversed[3]. However, the number of correct links in the Lo group tended to plateau (before the map was complete), while the number of incorrect links persisted and often increased in the latter portion of their learning activities.

In contrast, the Hi group generally added a few incorrect links but ultimately identified and removed them. These common patterns of map evolution suggest that students in the Lo group had two problems that the Hi group did not: (1) their monitoring skills were not strong enough to detect incorrect links, and (2) they

[2]The map score is defined as the number of correct links (based on the expert map) in the student's map minus the number of incorrect links.

[3]The only significant exceptions to the general monotonic trend in correct links were a few cases in which students deleted their entire map and started over. This occurred in students from both the Hi and Lo groups.

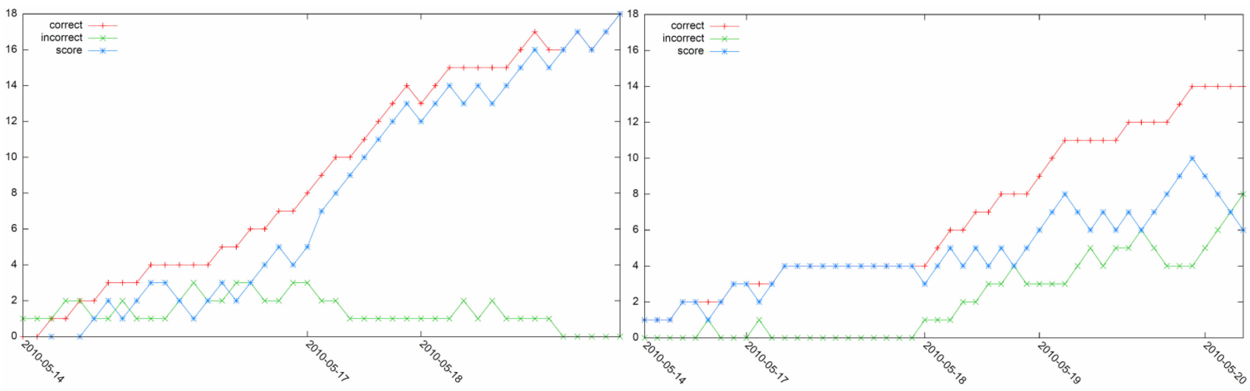

Fig. 1. Typical Map Evolution (from example Hi and Lo students)

did not seem to have the ability to read and comprehend the full set of resources to identify all of the correct links. These problems may have contributed to one another, and the sparsity of monitoring strategy feedback in this version of Betty's Brain was likely insufficient to help them overcome difficulties. To investigate these differences in performance and map evolution, we analyze students' interaction traces with our comparative data mining methodology, identifying and interpreting differences in learning behavior patterns.

3.2 Action Abstraction with Context

As with most computer-based learning environments, Betty's Brain produces extensive logs of the actions performed by students using the system. To effectively perform sequential data mining/analysis, raw logs of learning interactions must first be transformed into an appropriate sequence of actions by raising the level of abstraction from raw log events to a canonical set of distinct actions. In this step, researcher-identified categories of actions define an initial alphabet (set of action symbols) for the sequences. This also allows us to filter out irrelevant information (*e.g.*, cursor position) and to combine qualitatively similar actions (*e.g.*, querying an agent through different interfaces or about different concepts in a given topic).

In Betty's Brain we categorize student activities as five, qualitatively distinct actions: (1) *READ*: students access one of the pages in the resources; (2) *EDIT*: students add, delete, or modify a concept or link in the map; (3) *QUER*: students use a template to check their teaching by querying Betty, and she answers using causal reasoning through chains of links [Leelawong and Biswas 2008]; (4) *EXPL*: students probe Betty's reasoning by asking her to explain her answer to a query, which she does through a demonstration of her causal reasoning process; (5) *QUIZ*: students assess how well they have taught Betty by having her take a quiz, which is a set of questions chosen and graded by the Mentor agent.

Sometimes abstracting the raw log traces through action categorization also strips important context from the actions in the trace sequence. For example, while differentiating the addition of each possible link in a concept map would result in an unwieldy set of distinct actions, knowing that the link added is the same as one described in a previous read action or in a following query can be important context information. To maintain a balance between minimizing the number of distinct actions in the sequences and keeping relevant context information, we employ a metric indicating the relevance of each action to recent, previous actions. Based on this relevance metric we split each categorized action into two distinct actions: (1) high relevance and (2) low relevance to recent actions [Biswas et al. 2010].

For the case study, each student action was assigned a relevance score that depended on the number of relevant previous actions within a 3-action window. For Betty's Brain, a prior action is considered relevant to the current action if it is related to, or operates on, one of the same map concepts or links. This score provides a measure of *informedness* for map building/refinement actions and, similarly, a measure of *diagnosticity* for map monitoring activities [Biswas et al. 2010]. Overall, the relevance scoring maintains some action context by providing a rough measure of strategy focus or consistency over a sequence of actions.

3.3 Modeling Learning Behaviors with HMMs

Analyses that do not take into account the sequential nature of student interactions with the system, such as action frequency counts, capture limited, indirect information about learning strategies and students' cognitive state. HMMs provide a concise, higher-level representation of student learning strategies and behaviors (*e.g.*, strategies and their relationships, as opposed to simple action or sequence frequencies) [Jeong and Biswas 2008]. Previous work has illustrated some of the benefits of employing derived HMM states to represent learning strategies in analyzing student learning performance [Biswas et al. 2010].

Algorithms for learning an HMM from output sequences are well-known but require appropriate configuration/initialization parameters for effective use [Rabiner 1989]. Specifically, HMM learning algorithms start with an initial HMM whose parameters are iteratively modified to maximize the likelihood of producing observed output sequences. In particular, the number of states in the HMM and their initial output probabilities can have a profound effect on the resulting, learned HMM.

We have developed an algorithm that addresses these concerns in the construction of HMMs from a set of student activity sequences [Jeong and Biswas 2008; Biswas et al. 2010]. The hidden Markov model analysis in our methodology includes four major steps:

(1) *Model Initialization*: To determine the appropriate number of states for the HMM, our algorithm employs the Bayesian information criterion (BIC) [Heckerman et al. 1995], which balances a preference for concise models (*i.e.*, fewer states) with a preference for better-fitting models (*i.e.*, a greater likelihood of the model producing the observed activity sequences) [Li and Biswas 2002]. For each group of student interaction traces, we also employ a clustering algorithm to provide a set of state outputs for the respective group's initial HMM model. We cluster vectors of student activities (from each step in the student activity sequences), to find similar sets of activities at different points in the sequences. The resulting clusters of activities form the initial output probabilities for each state. This provides an initial model that helps explore a targeted portion of the possible model space.

(2) *Model Generation*: This is the core step in generating an HMM, with a variety of options in algorithms for learning HMMs from sequences. In this methodology, we employ the popular Baum-Welch algorithm [Baum et al. 1970] to the initial model and interaction traces to derive the optimal group HMM.

(3) *State Occurrence Calculation*: To determine the prevalence of individual states suggested by a generated HMM, we calculate the proportion of expected state occurrences [Biswas et al. 2010]. This metric employs the generated HMMs to calculate an expected value for the proportion of individual state occurrences. To maintain its relevance to the trace data and prevent the marginalization of states only occurring near the beginning of learning activities (as can happen with a stationary probability calculation), the expected value is calculated with lengths of student interaction traces in the group [Biswas et al. 2010].

(4) *Model Interpretation and Comparison*: The most difficult and subjective step in the process of analyzing student activity data through HMM generation is interpreting the resulting model. In this step, we assign meaning to the derived states of the model and generate behavior descriptions in terms of the interpreted states and their proportion of expected occurrences. In analyzing Betty's Brain interaction traces, we have found that the states in generated HMMs usually have very high self-loop probabilities (70% to 95% in most cases). This property of the HMMs reduces the complexity of assessing learning behaviors because each state can reasonably be considered to represent a learning behavior or similar set of learning behaviors. Finally, we compare the HMMs for the two student groups based on presence or absence of identified learning behaviors, as well as differences in their proportion of expected occurrences.

We employed this HMM generation and analysis to identify learning behaviors in the Hi and Lo map performance groups. The resulting HMMs for the two groups each had 4 states. Interestingly, at this level of analysis the learning behaviors appeared nearly identical. Based on state action emission probabilities in Table I, and using the methods detailed in [Biswas et al. 2010], we interpreted the states as representing:

—*Reading* - students are primarily engaged in reading the resources.

—*Informed editing* - students are primarily making informed edits related to recent activities, such as editing one area of the map at a time and linking their editing to recent reading, querying, and quizzing.

—*Uninformed editing* - students are primarily making unfocused or uninformed changes to their map, possibly indicating the use of trial-and-error and guessing strategies.

—*Checking* - students are querying and quizzing Betty to check the correctness of their causal maps and are making informed changes to improve their map. This state likely corresponds to the use of monitoring strategies that are encouraged by some of the Mentor agent feedback.

Table I. State Emission Probabilities

State	EDIT-H	EDIT-L	EXPL-H	EXPL-L	QUER-H	QUER-L	QUIZ-H	QUIZ-L	READ
Reading (Hi)	0%	0%	0%	0%	0%	0%	1%	0%	99%
Reading (Lo)	0%	0%	0%	0%	0%	0%	0%	0%	100%
Informed Editing (Hi)	89%	0%	0%	0%	8%	0%	3%	0%	0%
Informed Editing (Lo)	86%	0%	0%	0%	4%	0%	10%	0%	0%
Uninformed Editing (Hi)	3%	68%	0%	2%	2%	17%	1%	5%	2%
Uninformed Editing (Lo)	3%	69%	0%	4%	2%	11%	1%	7%	2%
Checking (Hi)	34%	4%	2%	0%	25%	2%	14%	11%	8%
Checking (Lo)	44%	1%	2%	0%	21%	1%	13%	11%	6%

The proportion of expected state occurrences for both HMMs indicated that students spent a little over half of their time in the Reading state and split the remainder of their time among the other three states. The only notable difference was a much higher likelihood in the Hi HMM of starting in the Reading state (86%) instead of the Uninformed editing state (14%), while the Lo group HMM had an equal split (50% and 50%) in the probability of starting in either of those states. Although the HMM analysis identified important learning behaviors common to both groups (and one noteable difference in starting behavior), a more fine-grained analysis is required to analyze the behavior differences between these two groups.

3.4 Comparative Sequence Mining

In many cases, the learning behavior patterns and strategies identified by comparison of HMMs can illustrate marked differences between groups of students. However, these models of learning behavior can also be too coarse-grained for some analyses, as illustrated by the HMM results for the Hi and Lo groups in Section 3.3. For example, a monitoring behavior might involve map editing, querying, explanation, and quizzing actions. The corresponding HMM state (with a high self-loop probability) would indicate what proportion of each activity was used, but not the order in which those actions occur. One monitoring strategy that would correspond to this state might be the use of regular queries to check recent map additions/changes against a student's current knowledge with periodic assessment of overall progress using a quiz. However, an equally applicable monitoring behavior for this state might be periodically taking a quiz and then using queries and explanations based on quiz questions to isolate incorrect information in the map.

This issue becomes even more pronounced when trying to compare across groups of students. When the HMMs generated for two groups contain similar states, it is still possible that those states represent different, although often related, learning strategies. To provide a finer-grained, complementary level of analysis, our methodology employs a novel combination of sequence mining techniques. In comparing across groups of students, the differences between the groups provide a natural criteria for identifying potentially important patterns. To use this criteria for mining important frequent patterns, we must define the appropriate measure(s) of frequency and the relevant difference calculated across the groups. The sequential pattern mining frequency measure (*i.e.*, how many students exhibit the given pattern) is important for identifying patterns common to a group. We refer to this as the "sequence support" (*s-support*) of the pattern, following the convention of [Lo et al. 2008], and we call patterns meeting a given s-support threshold *s-frequent*.

Another important metric for comparing patterns across groups is the episode frequency (*i.e.*, the frequency with which the pattern is repeated *within* an interaction trace). For a given trace, we refer to this as the "instance support" (*i-support*), following [Lo et al. 2008], and we call patterns meeting a given i-support threshold *i-frequent*. To calculate the i-support of a pattern in a *group* of traces, we use the mean of the pattern's i-support values across all traces in the group. Other definitions of group i-support could be employed (*e.g.*, a mean of i-support values normalized by the length of each trace or the maximum/minimum i-support of any trace in the group), but the average i-support provides a easily-interpreted value (*e.g.*, how frequently is the pattern employed by an "average" student representing the group) with a comprehensive view of pattern frequency for the group as a whole[4].

Our methodology combines the group s-support and i-support frequency measures to identify *differentially frequent* patterns across two groups of interaction traces. The first step in our comparative sequence mining technique is to use a sequential pattern mining algorithm to identify the patterns that meet a minimum s-support constraint within each group. However, another difficulty in mining frequent patterns in learning interaction traces is that the traces are "noisy." Specifically, students may exhibit a particular learning behavior pattern, but they may also perform additional actions interspersed with the actions of the pattern. Therefore, we employ a maximum gap constraint in mining a group's (s-support) frequent patterns. This means that between each consecutive pair of actions in a frequent pattern, the mining algorithm allows up to the maximum-gap number of additional actions.

Further, because this is an exploratory methodology, we employ a sequential pattern mining algorithm that allows regular expression constraints on the matched patterns. Specifically, we use the core algorithm from Pex-SPAM [Ho et al. 2005], which extends the fast SPAM algorithm [Ayres et al. 2002] with gap and regular expression constraints. Incorporating regular expression constraints provides the ability to focus the comparison between groups on patterns including a specific action, sub-sequence of actions, or any other pattern of interest expressed as a regular expression.

To compare the identified frequent patterns across groups, we calculate the difference in group i-support frequencies of each pattern. This focuses the comparison on patterns that are employed significantly more often by one group than the other. Again, to allow for noise in learning traces, we use a maximum gap constraint in calculating the component i-support values for each trace. Specifically, we calculate the individual i-support values as the maximum number of non-overlapping matches for the pattern (as defined for episode mining in [Laxman et al. 2005]) in an interaction trace, allowing up to the maximum-gap number of actions between consecutive pattern actions during matching.

This comparison produces four distinct categories of frequent patterns: two categories where the patterns are s-frequent in only one group, illustrating patterns primarily employed by the respective groups, and two categories where the patterns are common to both groups but used more often in one group than the other. The patterns in each of these qualitatively distinct categories are (separately) ranked by the difference in group i-support[5] to focus the analysis on the most differentially frequent patterns.

[4]In Betty's Brain interaction traces we have found similar results using both the mean i-support value and the mean normalized i-support value for a group.

[5]Even though a pattern may not be s-frequent in a group of interaction traces, it can still occur in some traces in the group, so an i-support value can be calculated (or the i-support is 0 if the pattern does not occur in any trace in the group).

Applying this comparison to the Hi and Lo map performance groups, we immediately noticed that many of the frequent patterns differed only by the number of consecutive reads or edits in the pattern. To improve this exploratory analysis, we revised the action abstraction transformation to distinguish a single action from repeated actions, which were condensed to a single "action" with the "-MULT" identifier. The average length of the transformed sequences was 124 actions with a standard deviation of 43.8 actions. Using the transformed sequences, our comparative sequence mining technique allowed us to identify a trend that was not apparent from separately considering s-frequent or i-frequent patterns, which were largely the same between the two groups. To illustrate these results, Table II presents the top three patterns (of length three) in each category[6]. In this analysis, we employed an s-support threshold of 50% to analyze patterns that were evident in the majority of either group of students.

The analysis illustrated that many of the patterns differing significantly between the groups showed the same trend: the Lo map performance group had a differential tendency to rely on knowledge construction sequences with multiple reads and multiple edits in a row, while the Hi group had a greater tendency to perform combinations of single read and edit actions.

Table II. Differentially Frequent 3-Action Patterns

Learning Activity Pattern	i-support (Hi - Lo)	s-frequent group	Hi group trend
READ $\prec$ EDIT-H $\prec$ READ	3.41	Hi	Hi group has greater tendency to use (READ $\prec$ EDIT)+
EDIT-H-MULT $\prec$ READ-MULT $\prec$ READ	2.59	Hi	
EDIT-H $\prec$ READ $\prec$ READ	2.50	Hi	
EDIT-H $\prec$ EDIT-H $\prec$ READ	2.38	Both	
EDIT-H $\prec$ EDIT-H $\prec$ READ-MULT	2.11	Both	
EDIT-H $\prec$ READ-MULT $\prec$ READ	1.92	Both	

Learning Activity Pattern	i-support (Hi - Lo)	s-frequent group	Lo group trend
READ-MULT $\prec$ EDIT-H-MULT $\prec$ EDIT-L-MULT	-0.77	Both	Lo group has greater tendency to use (READ-MULT $\prec$ EDIT-MULT)+
EDIT-L $\prec$ READ-MULT $\prec$ EDIT-H	-1.70	Both	
EDIT-L-MULT $\prec$ READ-MULT $\prec$ EDIT-H	-2.09	Both	
READ-MULT $\prec$ EDIT-H-MULT $\prec$ QUER-H	-1.70	Lo	
EDIT-L-MULT $\prec$ EDIT-H $\prec$ READ	-1.70	Lo	
EDIT-L $\prec$ READ-MULT $\prec$ EDIT-H-MULT	-1.80	Lo	

There are a number of possible explanations for the observed behavior differences between the groups. For example, the Hi performance students may have employed a knowledge construction strategy of reading a small portion of the resources and adding a link to their map each time they identified a causal relationship. The Lo performance students may have found it harder to find the causal information about links when reading. As a result they added incorrect links, and to correct these they did multiple reads, and then made one or more changes to the map that seemed to be based on guesswork rather than information obtained from the resources. This hypothesis is supported by the initial monotonic increase then plateauing of the number of correct links in the Lo students, as illustrated in Figure 1. In addition, they often added incorrect links that they could not remove because they did not find the right information while reading and/or they lacked monitoring strategies. Although our current comparative sequence mining analysis does not conclusively establish these hypotheses, it identifies trends that will be the basis of future investigation. Further, it suggests a future extension to this technique of incorporating performance measures related to actions (*e.g.*, edits) in the patterns.

[6]Although patterns of a variety of lengths were investigated, we only present the three-action patterns because the longer patterns tended to be supersequences of these patterns. The longer patterns illustrated the same trends, but did not elucidate any additional trends in the difference between Hi and Lo learning behaviors.

Table III. Differentially Frequent 3-Action (Query) Patterns

Learning Activity Pattern	i-support (Hi - Lo)	s-frequent group	Hi group trend
EDIT-H $\prec$ QUER-H $\prec$ EDIT-H	2.41	Hi	Hi group has greater tendency to use queries during reading
READ-MULT $\prec$ READ-MULT $\prec$ QUER-H	2.00	Hi	
QUIZ-H $\prec$ READ-MULT $\prec$ QUER-H	1.14	Hi	
READ-MULT $\prec$ EDIT-H $\prec$ QUER-H	1.47	Both	
READ-MULT $\prec$ QUER-H $\prec$ READ-MULT	1.27	Both	
QUER-H $\prec$ READ-MULT $\prec$ EDIT-H	1.25	Both	

Learning Activity Pattern	i-support (Hi - Lo)	s-frequent group	Lo group trend
QUIZ-H $\prec$ EDIT-H $\prec$ QUER-H	-0.19	Both	Lo group has greater tendency to use queries during editing
QUER-L $\prec$ READ-MULT $\prec$ EDIT-H	-0.45	Both	
EDIT-H-MULT $\prec$ QUER-H $\prec$ READ-MULT	-1.00	Lo	
QUER-H $\prec$ EDIT-L $\prec$ EDIT-H	-1.30	Lo	
READ-MULT $\prec$ EDIT-H-MULT $\prec$ QUER-H	-1.70	Lo	

Since the large majority of actions performed by students working with Betty's Brain were reading and map editing, the other actions are largely drowned out in the frequent patterns. Therefore, we extended our comparative sequence mining analysis by using a regular expression constraint to focus on monitoring patterns. For this case study, we focused on patterns including Query actions, which are important for effective monitoring of the knowledge construction and map building process. Table III presents the top three actions (again illustrated with three-action patterns[7]) in each category. The more frequent query patterns from the Hi group tended to include reading actions before and after queries, while the Lo group tended to use queries between editing actions. One interpretation of this difference is that the Lo performance students differentially used queries to monitor their changes to the map, while the Hi performance students used queries to ensure that their current map matched recently acquired information about chains of cause and effect. This result suggests an interesting focus for further investigation on the relationship between using queries to check map edits versus queries to check or understand the relationship between newly acquired knowledge and existing knowledge. Further, this behavior difference could have contributed to the difference in monitoring efficacy illustrated by the incorrect link trends in Figure 1.

4. DISCUSSION AND CONCLUSIONS

In this paper we presented an exploratory data mining methodology for analyzing learning behaviors from students' learning interaction traces. The integrated methodology combines action abstraction, HMM generation, and a novel sequence mining technique for comparative analysis of learning behaviors across student groups. We extended the HMM analysis, which identifies overall learning behaviors common to a group of students, with a complementary sequence mining technique to identify differentially frequent patterns between the groups. Results from a recent classroom study with Betty's Brain illustrate the effectiveness of this integrated methodology for: (1) pre-processing logs by categorizing actions and maintaining action context with a relevance metric, (2) identification of knowledge construction and monitoring behaviors in group HMMs for a class of students categorized by performance and evolution of their causal concept maps, and (3) identification and comparison of additional trends in group learning behaviors through analysis of differentially frequent patterns in student interaction traces.

In future work, we intend to expand upon this research through a variety of enhancements and additional applications. We will incorporate map evolution (*e.g.*, whether an edit resulted in a better or worse map) into the differentially frequent pattern analysis to provide a direct link between identified behaviors and performance. Further, we are working on creating a library of interaction trace segments that are representative of identified learning strategies for use in online HMM analysis of student activities. Online analysis to

[7]Note: there were only two of the more i-frequent Lo patterns that were s-frequent in both groups.

categorize student activities by comparison with this HMM library can facilitate more targeted scaffolding and feedback in computer-based learning environments, such as Betty's Brain. Finally, we intend to more tightly integrate the sequence mining and HMM analyses by marking frequent patterns in interaction traces and building HMMs in which short patterns are the emissions of each state instead of individual actions. This integration will enhance the methodology with a more detailed representation of group learning behaviors and strategies.

REFERENCES

Agrawal, R., Imieliński, T., and Swami, A. 1993. Mining association rules between sets of items in large databases. In *Proceedings of The ACM SIGMOD International Conference on Management of Data*. Vol. 22. ACM, 207–216.

Agrawal, R. and Srikant, R. 1995. Mining sequential patterns. In *Proceedings of the Eleventh IEEE International Conference on Data Engineering (ICDE)*. 3–14.

Ayres, J., Flannick, J., Gehrke, J., and Yiu, T. 2002. Sequential pattern mining using a bitmap representation. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, 429–435.

Baum, L., Petrie, T., Soules, G., and Weiss, N. 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics 41,* 1, 164–171.

Biswas, G., Jeong, H., Kinnebrew, J., Sulcer, B., and Roscoe, R. 2010. Measuring Self-regulated Learning Skills through Social Interactions in a Teachable Agent Environment. *Research and Practice in Technology-Enhanced Learning (RPTEL) 5,* 2, 123–152.

Biswas, G., Leelawong, K., Schwartz, D., Vye, N., and Vanderbilt, T. 2005. Learning by teaching: A new agent paradigm for educational software. *Applied Artificial Intelligence 19,* 3, 363–392.

Boyer, K., Ha, E., Wallis, M., Phillips, R., Vouk, M., and Lester, J. 2009. Discovering Tutorial Dialogue Strategies with Hidden Markov Models. In *Proceedings of the 14th International Conference on Artificial Intelligence in Education, Brighton, UK*. IOS Press, 141–148.

Cohen, P. and Beal, C. 2009. Temporal data mining for educational applications. *Int J Software Informatics 3,* 1.

Ding, B., Lo, D., Han, J., and Khoo, S. 2009. Efficient mining of closed repetitive gapped subsequences from a sequence database. In *Proceedings of the IEEE International Conference on Data Engineering*. IEEE, 1024–1035.

Heckerman, D., Geiger, D., and Chickering, D. M. 1995. A tutorial on learning with Bayesian networks. *Machine Learning 20*, 197–243.

Ho, J., Lukov, L., and Chawla, S. 2005. Sequential pattern mining with constraints on large protein databases. In *Proceedings of the 12th International Conference on Management of Data (COMAD)*. 89–100.

Jeong, H. and Biswas, G. 2008. Mining student behavior models in learning-by-teaching environments. In *Proceedings of The First International Conference on Educational Data Mining*. Montreal, Quebec, Canada, 127–136.

Laxman, S., Sastry, P., and Unnikrishnan, K. 2005. Discovering frequent episodes and learning hidden markov models: A formal connection. *IEEE Transactions on Knowledge and Data Engineering*, 1505–1517.

Leelawong, K. and Biswas, G. 2008. Designing learning by teaching agents: The Betty's Brain system. *International Journal of Artificial Intelligence in Education 18,* 3, 181–208.

Li, C. and Biswas, G. 2002. A Bayesian Approach for Learning Hidden Markov Models from Data. *Scientific Programming: Special Issue on Markov Chain and Hidden Markov Models 10,* 3, 201–219.

Lo, D., Khoo, S., and Liu, C. 2008. Efficient mining of recurrent rules from a sequence database. In *Proceedings of the 13th International Conference on Database Systems for Advanced Applications*. Springer-Verlag, 67–83.

Mannila, H., Toivonen, H., and Inkeri Verkamo, A. 1997. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery 1,* 3, 259–289.

Pardos, Z. and Heffernan, N. 2011. Using HMMs and bagged decision trees to leverage rich features of user and skill from an intelligent tutoring system dataset. *To Appear in the Journal of Machine Learning Research (JMLR)*.

Rabiner, L. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE 77,* 2, 257–286.

Shih, B., Koedinger, K., and Scheines, R. 2010. Unsupervised Discovery of Student Learning Tactics. In *Proceedings of the Third International Conference on Educational Data Mining.*

Wagster, J., Tan, J., Wu, Y., Biswas, G., and Schwartz, D. 2007. Do learning by teaching environments with metacognitive support help students develop better learning behaviors. In *Proceedings of the 29th Annual Meeting of the Cognitive Science Society*. Nashville, TN, 695–700.