Learning Behavior Characterization with Multi-Feature, Hierarchical Activity Sequences

Cheng Ye
Department of EECS and ISIS
Vanderbilt University
1025 16th Ave S, Ste 102
Nashville, TN 37212
cheng.ye@vanderbilt.edu

John S. Kinnebrew
Department of EECS and ISIS
Vanderbilt University
1025 16th Ave S, Ste 102
Nashville, TN 37212
john.s.kinnebrew@vanderbilt.edu

James R. Segedy
Department of EECS and ISIS
Vanderbilt University
1025 16th Ave S, Ste 102
Nashville, TN 37212
james.segedy@vanderbilt.edu

Gautam Biswas
Department of EECS and ISIS
Vanderbilt University
1025 16th Ave S, Ste 102
Nashville, TN 37212
gautam.biswas@vanderbilt.edu

ABSTRACT

This paper discusses Multi-Feature Hierarchical Sequential PAttern Mining, MFH-SPAM, a novel algorithm that efficiently extracts patterns from students' learning activity sequences. This algorithm extends an existing sequential pattern mining algorithm by dynamically selecting the level of specificity for hierarchically-defined features individually for each pattern. Consequently, MFH-SPAM operates on a larger space of patterns in the activity sequences. In this paper, we employ a differential version of MFH-SPAM to extract a small set of patterns that best differentiate students with different learning behavior profiles in the Betty's Brain system. Our results illustrate that: (1) MFH-SPAM identifies important patterns missed by traditional sequence mining approaches; and (2) the differential patterns provide additional information for characterizing learning behaviors. This has implications for developing targeted and adaptive scaffolding in open-ended learning environments.

1. INTRODUCTION

Open-Ended Learning Environments (OELEs [4,7]) present students with a challenging problem-solving task, along with resources and tools for solving the task. Students have the choice to explore, and, therefore, can evolve their solutions in a variety of ways. In previous work, we proposed a theory-based approach called *coherence analysis* (CA) [7] for analyzing student behavior in OELEs. Experimental results showed that grouping students using the CA metrics produced distinct behavior profiles that are discussed in greater detail in Sections 3 and 4. To date we have established the stability and usefulness of our CA measures across extended periods of student work, which does not make this approach directly applicable to adaptive scaffolding as students work in the OELE. To address this problem, our goal has been to use sequence mining methods to find students' activity patterns that are indicators of their behavior profiles. In this paper, we present a case study illustrating that action patterns derived using a novel hierarchical sequence mining approach followed by differential analysis enable classification performance on a par with the groupings derived using CA. Occurrence of individual action patterns can be easily detected online, and future work will assess their utility for early identification of behavior profiles and contextualized scaffolding in OELEs.

In the Betty's Brain OELE [5] each action performed by a student has a number of accompanying features that capture context and consequences of the action. In past work, we used pre-processing methods to select specific features and the level of granularity for each feature to generate 'flat' sequences for pattern mining [2]. This largely ad hoc process resulted in our running many different mining analyses, but often missing potentially important patterns. Other work, such as Plantevit et al. [6], has addressed some aspects of the search in large feature spaces. They define a two-phase technique that first determines frequent combinations of features and levels of specificity in hierarchical representations to pre-processes multi-feature (hierarchical) sequences into a 'flattened' representation. While this approach provides clear advantages over numerous mining analyses with ad hoc feature and granularity choices, many frequent patterns can still be missed due to the initial flattening phase.To address this issue, we have developed a novel **M**ulti-**F**eature, **H**ierarchical **S**equential **PA**ttern **M**ining algorithm (MFH-SPAM).

MFH-SPAM extends the sequence mining algorithm SPAM [1] to simultaneously operate on the entire feature space of action sequences for pattern mining. In this work, we start with MFH-SPAM, and then apply a classifier wrapper method [3] to discover a small subset of mined patterns that are useful for differentiating students across the CA-derived learning behavior profiles. We have evaluated MFH-SPAM and other traditional sequence mining approaches in this behavior profile classification task using data from a recent study with the *Betty's Brain* OELE. Results show that MFH-SPAM consistently outperforms traditional sequence mining approaches on this task. Further, the differential patterns provide additional information for characterizing student learning behaviors, which has implications for developing targeted and adaptive scaffolding in OELEs.

2. MFH-SPAM APPROACH

Our approach to efficient mining of Multi-Feature, Hierarchical (MFH) sequences extends the SPAM algorithm [1] by directly working with the MFH representation of actions during the mining process. To illustrate this representation, we consider a generic set of possible items/actions to make up sequences (A, B, or C) with an additional feature (*e.g.*, a measure of the action's outcome) that can take on values of $+$ or $-$ at the most general level. In this example, $+$ values for the outcome feature can be further specified as either $+Big$ or $+Small$ at the next level of the hierarchy. Therefore, an individual action might be represented as $B^{+\mathrm{Big}}$, and both $B^{+\mathrm{Big}}$ and $B^{+\mathrm{Small}}$ actions could be more generally represented as B^{+} by abstracting the outcome feature to the more general $+$ level. Further, $B^{+\mathrm{Big}}$, $B^{+\mathrm{Small}}$, and B^{-} actions could all be represented as simply a B action by ignoring the outcome feature entirely. We represent one action followed by another in a sequential pattern using the $\rightarrow$ symbol, such as $A \rightarrow B$ to indicate A followed by B. Itemsets (*i.e.*, co-occurring items in the sequence) are surrounded with parentheses, such as (A, B) to indicate both A and B occurring at the same position in a sequence (*i.e.*, simultaneously).

The core SPAM [1] algorithm searches the space of possible sequential patterns by incrementally extending the current pattern (starting with an empty pattern) in a depth-first manner. For each pattern in the search, SPAM generates the potential "child" patterns by applying one of two types of extensions to the current pattern: 1) a Sequence-extension step (S-step), which appends an item to the end of the sequence (occurring after the last item/itemset), or 2) an Itemset-extension step (I-step), which adds an additional item to the last itemset in the current pattern. For each pattern considered, SPAM calculates the number of sequences in which the pattern occurs using a vertical bitmap representation, explained in more detail later. If the number of sequences in which the new pattern is contained is less than the specified support threshold, SPAM rejects the pattern and does not consider any subsequent extensions to it.

MFH-SPAM augments SPAM with two new pattern extension steps in the pattern search: Feature extensions (F-steps) and Hierarchical extensions (H-steps). During an F-step, MFH-SPAM adds an additional feature to the last item of the current sequence using one of the most general values in the feature hierarchy. For example, the possible extensions to the pattern $A \rightarrow B$ with an F-step would result in $A \rightarrow B^{+}$ or $A \rightarrow B^{-}$. During an H-step, MFH-SPAM selects the last feature of the last item of the current sequence and specifies its value at one level deeper in the feature hierarchy. For example, the possible extensions to the pattern $A \rightarrow B^{+}$ with an H-step would result in $A \rightarrow B^{+\mathrm{Big}}$ or $A \rightarrow B^{+\mathrm{Small}}$.

In addition to these two new extension steps in MFH-SPAM, we define a corresponding extension to the vertical bitmap approach employed in SPAM to efficiently calculate the support for a new pattern[1]. For each data sequence, SPAM initially defines a bitmap for each possible item (*e.g.*, A, B, and C) that represents the locations of that item in the sequence with a value of 1 (all other locations have a value of 0). For example, the sequence $A \rightarrow B \rightarrow B$ would be represented with an A bitmap of [1 0 0], a B bitmap of [0 1 1], and a C bitmap of [0 0 0]. As SPAM generates patterns, it combines item bitmaps to produce *pattern bitmaps* in which 1's represent the endpoints of the corresponding pattern in the sequences. Consequently, for a trivial, single-item pattern like A, the pattern bitmap is exactly the same as the initial item bitmap.

For an S-step extension of a pattern (*e.g.*, extending A to $A \rightarrow B$), SPAM first transforms the current pattern bitmap ([1 0 0]) to indicate where the extension to the current pattern could occur. This is performed by shifting the bitmap to make each location following the occurrence of a 1 in the pattern bitmap a 1 (indicating a *candidate location* for the additional item being added in the S-step) and making all other locations 0 (e.g., resulting in the bitmap [0 1 0]). In other words, $A \rightarrow B$ exists in the sequence if B exists in the candidate location of the second position in the sequence. To complete the S-step (*e.g.*, for A to $A \rightarrow B$) SPAM performs a bitwise AND operation on the transformed pattern bitmap and the item (B) bitmap, resulting in the new pattern bitmap of [0 1 0] indicating that the pattern $A \rightarrow B$ exists and ends at the second position in the sequence.

We extend the SPAM bitmap procedure in F- and H-steps by first creating bitmaps for each possible feature value (at every level of the hierarchy) in the sequence, just as SPAM does with each possible item. Thus, if the original sequence were $A^{-} \rightarrow B^{+\mathrm{Big}} \rightarrow B^{+\mathrm{Small}}$, we would have a $-$ bitmap of [1 0 0], a $+$ bitmap of [0 1 1], a $+Big$ bitmap of [0 1 0], and a $+Small$ bitmap of [0 0 1]. The bitmap operations for F- and H-steps are then analogous to those for S-steps except without the bitmap shift[2] and using the feature value bitmap corresponding to the chosen extension. For example, applying an F-step to add the outcome feature with a value of $+$ to the pattern $A \rightarrow B$, producing $A \rightarrow B^{+}$, would correspond to [0 1 0] (the pattern bitmap) AND [0 1 1] (the feature value bitmap), giving the new pattern bitmap [0 1 0], indicating that this pattern does occur in the example sequence and ends at the second position in the sequence. With the additional F- and H-steps, as well as corresponding bitmap operations for calculating support, MFH-SPAM extends SPAM to efficiently search the space of possible patterns in MFH sequences. Finally, to choose a small subset of the frequent patterns identified by MFH-SPAM (or by SPAM for the experimental comparison) that differentiate the pre-defined learning profiles, we apply a classifier wrap-

[1]In the algorithm description, we describe only the case in which no gaps are allowed between items in the pattern, however, implementing more general gap constraints works in the same manner as with extensions to the original SPAM algorithm

[2]No shift is necessary because the candidate location is for adding further detail to the last item in the current pattern rather than adding an item after it.

per method [3]. Using a greedy approach, the classifier wrapper iteratively identifies the best pattern to include next[3].

3. DATA AND EVALUATION METHODS

The data presented in this paper comes from a study of 98 students from four middle school science classrooms using *Betty's Brain* for six weeks [7]. Six coherence measures were employed to describe the quality and quantity of various problem-solving activities for each student, and hierarchical clustering with these measures identified three primary clusters of students characterized by different behavior profiles [7]. In total, 87 of the students fell into one of these three clusters, and the other 11 students exhibited behavior profiles indicative of either extreme confusion or disengagement. The primary clusters were defined as:(1) *Frequent researchers and careful editors*, who spent large proportions of their time viewing sources of information and did not edit their maps very often; (2) *Strategic experimenters*, who spent a fair proportion of their time viewing sources of information, but often did not take advantage of this information; and (3) *Engaged & efficient students*, who edited their maps very frequently, and usually supported by information from previous activities.

To generate MFH activity sequences for mining, we categorized learning actions into seven primary categories, defined hierarchically (these categories are discussed in more detail in [2]): *Read*ing resource pages; *Search*ing the resources for keywords; causal *Map Edit*ing; *Query*ing the teachable agent, Betty; having Betty take a *Quiz*; asking Betty to *Explain* her answer; or taking *Note*s or causal link annotations (*LinkEval*) indicating whether a link is believed to be correct. To capture the context associated with these actions, we use additional features: (1) the "Length" dimension (applied to Read actions) indicates whether the student spent enough time on the page to have read a significant amount of the material (Full) or only spent a brief period of time on the page (Short) [2]; (2) the "Previous (Full) Read" dimension indicates whether the student has previously done an in-depth ("Full") read of the page or not; (3) the "Supported" dimension indicates whether or not an EditLink action was based on either recently viewed reading materials or quiz results [7], with supported actions denoted by *Sup* and unsupported actions denoted by *NoSup*; and (4) the "Map Score Change" dimension indicates what effect an EditLink action had on the quality of the student's map - whether the quality improved (denoted by +), worsened (denoted by −), or did not change (denoted by =).

We evaluate our MFH-SPAM approach with comparison to four alternative approaches: *Flattened Features (SPAM)* first flattened all activity sequences using all features and the greatest level of action specificity and then used SPAM to generate candidate patterns (e.g., this approach would consider the pattern $\text{LinkRem}^{+}_{\text{Sup}} \rightarrow \text{LinkAdd}^{-}_{\text{NoSup}}$, but it would not consider the more general pattern LinkEdit → LinkEdit); *Actions-only (SPAM)* considered only the frequent patterns at the most general level of specificity and did not consider any additional features; *MFH-SPAM Baseline by Frequency* used our MFH-SPAM algorithm to generate candidate patterns and simply selected the 10 most frequent patterns; and *Coherence Metrics* classified students using the coherence measures. The performance of each approach was evaluated as the average F1 score of the resulting classifier using 10-fold cross-validation. We chose decision trees as the classifiers and performed this analysis at mining support thresholds ranging from 1.0 to 0.5 in increments of 0.02.

[3] A limit of 10 patterns and an increase of at least 0.1% in performance over the previous pattern set was used in our implementation of the wrapper. A stratified 5-fold cross-validation approach was used for building the classifier in the wrapper with F1 score for evaluation.

4. RESULTS

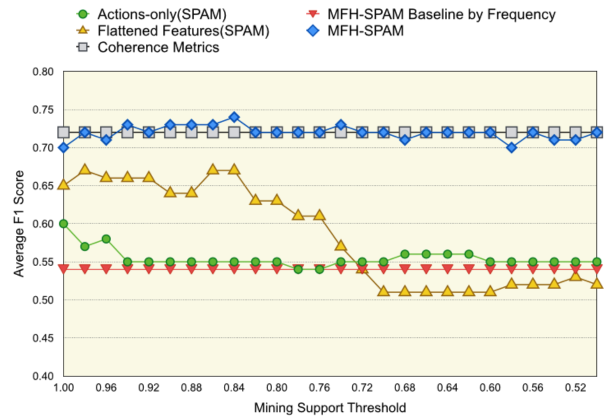

Figure 1: Classification performances of MFH-SPAM and alternative approaches

Figure 1 illustrates the performances of the classifiers built using the candidate feature sets mined in each approach. At each level of support, MFH-SPAM achieved an average F1 score that was much higher than the scores produced by the other sequence mining methods. When using a particularly high mining support threshold, the Flattened Features approach achieves performance close to that of MFH-SPAM, but its performance decreases dramatically as the support threshold is reduced (and the search space is increased). One striking result from this analysis is that MFH-SPAM's performance is on par with the performance of the classifier trained with the features used to perform the original clustering that defined these behavior profile classes. Further, Table 1 presents the five patterns chosen most frequently across the 10 cross-validation folds at a support threshold of 0.9. Considering these top patterns, it is clear that the first three patterns could not have been identified without MFH-SPAM, as they involve multiple levels of hierarchies and feature specificity.

Interestingly, the top MFH-SPAM patterns all involve various forms of causal link edits. This suggests that the way a student went about building their map, as opposed to the way they navigated the resources and investigated Betty's quiz results, was the most useful in predicting their overall learning behavior profile. However, the edit actions, through the support feature, can also incorporate the action's relationship to reading and quiz actions. In other words, what was most helpful in predicting a student's cluster was not the way they acquired information (either from the resources or quiz results), but how they applied previously acquired information to editing their maps. When comparing frequency of use across the three groups, their relative magnitudes are

Table 1: **Pattern Frequency Mean (Std Dev) by Cluster for MFH Wrapper with Support 0.9**

Pattern	Researchers	Experimenters	Efficient
$\text{LinkRem}^{+}_{\text{Sup}} \rightarrow \text{LinkEdit}$	2.6 (2.5)	3.6 (3.0)	14.3 (8.4)
$\text{LinkEdit}^{+}_{\text{Sup}} \rightarrow \text{LinkAdd}$	2.3 (1.9)	2.5 (2.6)	12.0 (6.5)
$\text{LinkEdit}^{+}_{\text{NoSup}} \rightarrow \text{LinkEdit}$	3.3 (2.9)	16.4 (16.9)	15.6 (12.3)
$\text{LinkEdit}^{-} \rightarrow \text{LinkEdit}^{-}$	3.7 (3.1)	17.5 (16.0)	18.3 (16.2)
LinkAdd^{-}	15.3 (7.2)	28.6 (12.1)	43.5 (21.6)

compatible with the behavior descriptions; *e.g.*, researchers and careful editors make the least number of these edits; engaged & efficient students have the most; and strategic experimenters fall in between. This confirms that the engaged & efficient students, who exhibited the best learning behaviors and the largest learning gains [7], are broadly distinguished from the other groups by more map editing overall: ineffective and effective; supported and unsupported. The usage distributions for these patterns also revealed interesting characteristics about strategic experimenters. These students performed patterns with supported edits far less frequently than engaged & efficient students. Conversely, they performed patterns with unsupported edits far more frequently than researchers and careful editors. Thus, even though the engaged & efficient students made several unsupported and ineffective edits, it would seem that their overall edit distribution is far more favorable to achieving better map scores (and in their case, better pre-post gains on domain knowledge) than that of the strategic experimenters.

To better characterize these three groups, we followed up on previous experimental results [2] and further analyzed the top behavior pattern: (1) $\textit{LinkRem}^{+}_{\text{Sup}} \rightarrow \textit{LinkEdit}$ that indicates an effective map correction behavior (removing an incorrect link with supporting evidence) followed by further editing. Overall, an average of 19% (s.d. 9%) of the engaged/efficient students' total number of link edits involved this pattern versus 9% (s.d. 8%) for researchers/careful-editors and 9% (s.d. 7%) for strategic experiments. This behavior of incorporating effective map correction in periods of extended map editing appears to be a key characteristic of the engaged/efficient students. Further analysis also suggested that engaged/efficient students were relatively more likely to follow this pattern with a quiz to evaluate their revised map than the researchers/careful-editors and strategic experimenters. This may indicate a greater propensity for the engaged/efficient students to effectively combine evaluation of the causal map with map construction and correction. In summary, going back to OELE characteristics, the engaged and efficient students seem to be better at exploring the problem-solving space, and in distinguishing correct and incorrect approaches to solving complex problems.

5. DISCUSSION AND CONCLUSIONS

MFH-SPAM provides a comprehensive approach to mining OELE activity sequences by efficiently covering the entire MFH action-feature space to generate patterns. Results showed that MFH-SPAM consistently outperforms traditional sequence mining approaches on a behavior profile classification task. Further, analysis of the MFH-SPAM patterns illustrated that a nice, compact way for differentiating these student groups, while retaining high accuracy, was in their approach to map construction and refinement using various forms of editing actions. Overall, these results showed the importance of behavior patterns identified by MFH-SPAM and illustrated the potential to use these patterns to better characterize and ultimately scaffold student learning. In general, effective virtual agents for adaptive scaffolding in OELEs like Betty's Brain may do well to focus on behavior patterns to gain an understanding of how students' *apply their acquired knowledge* (*e.g.*, from reading the resources and studying quiz results) to build and refine models. Detection of specific suboptimal (not using acquired information well) or erroneous behaviors in this context may provide the needed cue for effective scaffolding.

6. ACKNOWLEDGMENTS

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A120186 to Vanderbilt University. The opinions expressed are those of the authors and do not represent the views of the Institute or the U.S. Department of Education.

7. REFERENCES

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

[2] J. S. Kinnebrew, K. M. Loretz, and G. Biswas. A contextualized, differential sequence mining method to derive students' learning behavior patterns. *Journal of Educational Data Mining*, 5(1):190–219, 2013.

[3] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[4] S. Land. Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research and Development*, 48(3):61–78, 2000.

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

[6] M. Plantevit, A. Laurent, D. Laurent, M. Teisseire, and Y. W. Choong. Mining multidimensional and multilevel sequential patterns. *ACM Transactions on Knowledge Discovery from Data*, 4(1):4:1–4:37, Jan. 2010.

[7] J. R. Segedy, J. S. Kinnebrew, and G. Biswas. Using coherence analysis to characterize self-regulated learning behaviours in open-ended learning environments. *Journal of Learning Analytics*, To Appear.