CCVis: Visual Analytics of Student Online Learning Behaviors Using Course Clickstream Data

Maggie Celeste Goulden; Trinity College Dublin
Eric Gronda; University of Maryland, Baltimore County
Yurou Yang, Zihang Zhang; Zhejiang University
Jun Tao, Chaoli Wang, Xiaojing Duan, G. Alex Ambrose, Kevin Abbott, Patrick Miller; University of Notre Dame

Abstract

As more and more college classrooms utilize online platforms to facilitate teaching and learning activities, analyzing student online behaviors becomes increasingly important for instructors to effectively monitor and manage student progress and performance. In this paper, we present CCVis, a visual analytics tool for analyzing the course clickstream data and exploring student online learning behaviors. Targeting a large college introductory course with over two thousand student enrollments, our goal is to investigate student behavior patterns and discover the possible relationships between student clickstream behaviors and their course performance. We employ higher-order network and structural identity classification to enable visual analytics of behavior patterns from the massive clickstream data. CCVis includes four coordinated views (the behavior pattern, behavior breakdown, clickstream comparative, and grade distribution views) for user interaction and exploration. We demonstrate the effectiveness of CCVis through case studies along with an ad-hoc expert evaluation. Finally, we discuss the limitation and extension of this work.

Introduction

Higher education in the United States has been facing a great challenge of encouraging college students to complete their courses. The most recent report from the National Center for Education Statistics [17] shows that, in 2016, the retention rate (i.e., the percentage of students returning in the subsequent year) was 81%, and the six-year graduation rate for students entering in 2010 was only 60%. Further study suggests that the retention rate is related to student performance [3], and the first year is the most crucial one for students to complete their program [2]. To help students succeed in their course programs and encourage them to return for the subsequent years, it is critical for instructors to understand the learning behaviors of students, identify the ones at risk at the earliest possible stage, and help students improve their performance. It is also important for students to understand peer learning habits and improve their own.

Nowadays, the use of a Learning Management System (LMS), such as Sakai, Moodle, or Blackboard, is prevalent in college education. Such an LMS helps the instructor deliver material to the students, administer tests and other assignments, track student progress, and manage record-keeping. It also records student online learning activities and provides a new opportunity to understand their learning behaviors. However, in order to fully utilize this value we must overcome new challenges posed by the complexity of the recorded data. First, most of the integrated analysis tools in the LMS focus on simple statistics, which could be quite limited for in-depth analytics. For example, the number of activities of a student is often used to determine how active the student performs, which could be misleading. In our study, we find that "visiting homepage → idle → visiting homepage → idle" is one of the most common activity sequences. The activities in this sequence, however, are irrelevant to the study of the course but could significantly increase the number of activities performed by students. Second, effectively grouping students is critical for scalable analytics of introductory courses with a large number of enrollment. However, the current learning analytics tools often fail to group students based on their inherent learning behaviors. Students could often be clustered based on their assignment or performance, but using a more fundamental machine learning algorithm could reveal the hidden behavior patterns that play a key role in student performance.

We present the CCVis, Course Clickstream Visualization, a visual analytics tool for analyzing student course clickstream data in an LMS and exploring student online learning behaviors. CCVis goes beyond simple statistics by employing the advanced techniques including higher-order network and structural identity classification to analyze, categorize, and summarize student learning behaviors. We design coordinated multiple views (CMVs) to enable not only an effective overview of the massive clickstream data but also the detailed comparison of individual students' behaviors. The ultimate goal of CCVis is to facilitate instructors in monitoring and managing student progress and performance.

FYE Course and LMS Information

The course clickstream data were collected from the Moreau First Year Experience (FYE), which is a required, two-semester course sequence that helps first-year students make a meaningful transition to collegiate life at the University of Notre Dame. Each year, over 2,000+ students take FYE with 125 instructors. FYE is graded and carries one credit hour per semester, meeting 50 minutes each week for 13 weeks. It employs a flipped-classroom model: individual students preview and prepare online materials common to all sections (articles, videos, webpages, surveys, etc.) in advance of class and prepare short written prompts to launch small group discussions in class. Time in class focuses on student-centered activities that facilitate the integration of academic, co-curricular, and residential experiences.

FYE uses Sakai, an LMS for the overarching structure of the course. Each individual course site has the course syllabus and all student resource materials; for instructors there are additional links for suggested in-class activities and particular instructor resources. The Sakai site functions as follows: (1) students and instructors access the readings/viewings on a weekly basis; (2) prior to class students submit their brief weekly reflections in "quizzes"; (3) instructors evaluate work on a weekly basis and update "gradebook" based on common rubrics; (4) at midterm and final instructors add grades for participation; and (5) also at midterm and final students submit a multimedia ePortfolio assignments through an interface with "Digication".

Course Clickstream Data

The clickstream data set we use was collected from the FYE course taken by the first-year students during the spring semester of 2018. This course was divided into 114 sections to maintain the optimal student-instructor ratio. The content and assessment activities of the course were consistent across all the sections and delivered in the same learning environment, which allowed us to collect uniform data on the course activities of all students, resulting in 2.3 million click tracks. The activities collected for analysis include course logins, content clicks, and assignment submissions. The content includes reading and video materials. Each activity is presented in the format of "student *s* visited webpage *w* at time *t*". In addition to the activity data, we also collected and analyzed the performance indicator data such as Weekly Prompts (1-11) scores, ePortfolio Access and Link Check scores, Before/After Spring Break participation scores, Integration #3 scores, and Capstone ePortfolio scores. The analysis of this large data set has the potential of revealing the patterns of student learning style that may be overlooked in small classes.

Related Work

Visual analytics of temporal event sequences. Temporal event sequences have been extensively studied in previous work. To effectively extract and present information in event sequences, two major challenges are tackled: the *volume of data* and the *variety of patterns*. Du et al. [12] surveyed the methods for addressing both challenges. They described 15 strategies which fall into four groups: extraction, temporal folding, pattern simplification, and iterative strategies. Wongsuphasawat et al. [24] presented LifeFlow to aggregate multiple sequences based on the events. The aggregated sequences form a tree, preserving the common patterns. CMVs are used to display the detailed event information. Wongsuphasawat and Gotz [25] further developed Outflow as an extension to LifeFlow. Outflow allows repetition of events and aggregates the sequences to form a graph. Sankey diagram is used to visualize the aggregated sequences. Liu et al. [15] proposed CoreFlow to summarize the sequences based on core events. This approach extracts the core events and constructs a tree to encode their relationships. The branching structure of the tree describes the general patterns in the sequences. Bodesinsky et al. [5] designed a visual analytics system with CMVs to explore sequences of events. The event view visualizes individual sequences as horizontally aligned bars, where each bar represents an event. A pattern overview is used to summarize the common event patterns for users to query and highlight patterns of interests in the event view. Partl et al. [18] proposed Pathfinder to study paths in multivariate graphs. The interface consists of a node-link diagram to show the topology of paths and a ranked list to show the attributes associated with the nodes. Malik et al. [16] compared two groups of sequences using high-volume hypothesis testing. The statistics information of the same sequences in two groups is derived for users to visually compare the two groups. Chen et al. [9] leveraged the minimum description length principle to summarize the event sequences. The cost function considers both the total pattern lengths and the edit distance between the sequences and generated patterns. Steptoe et al. [21] converted user trajectories in theme parks into event sequences. Each event encodes the time spent on a certain location. The events are visualized as bars, whose color indicate the duration of time.

Unlike the existing approach, which usually summarized the common patterns in the sequences, our approach leverages a higher-order network construction algorithm to extract the critical sequences that lead to different transition probabilities. Additionally, higher-order network synthesizes the relationships among these sequences, allowing large-scale features to be studied.

Learning analytics. Hsieh and Wang [14] proposed a data mining approach to construct a learning path using formal concept analysis and recommended learning objects using both preference-based and correlation-based algorithms. However, their recommendation mainly depends on the content of materials, and the students' behaviors and learning habits are less considered. Arnold et al. [1] developed the Course Signals to allow faculty members to provide each student real-time feedback via a personalized email, as well as a specific color on a stoplight—traffic signal. Charleer et al. [7] developed a learning analytics dashboard called LISSA to facilitate communication between advisors and students by visualizing grade data. Derick et al. [10] developed AffectVis to visualize learner's affect states and show their connection with specific learning activities.

In addition to LMS, the visual analytics approach has been developed for massive open online courses (MOOC) to study interactions among users and learning behaviors of students. Trimm et al. [23] visualized the students' progress trajectories over semesters. The trajectories of multiple students are clustered and composited for visualizing the performance of student groups. Dernoncourt et al. [11] presented MoocViz, which provides a cross-platform data analytics framework for researchers to embed additional modules. Learning statistics of students from different countries for multiple courses are visualized as a demonstration. Shi et al. [20] proposed VisMOOC to explore MOOC video clickstream data. VisMOOC visualizes the temporal variation of different types of clicks as a stacked bar chart, and the forward and backward seek events using parallel coordinates. Wu et al. [28] developed NetworkSeer to understand the interactions among students. A parallel coordinates view shows multiple properties of students for users to group and filter the students according to these properties. The interactions are depicted by a node-link diagram. Chen et al. [8] designed PeakVizor to study the "peaks" in MOOC video clickstream data. Each peak is visually encoded by a glyph in an overview. The spatial-temporal information of the peaks and the correlation between the peaks are visualized in two additional views. Fu et al. [13] proposed iForum, a visual analytics system to understand the activities on MOOC forums. CMVs are used to present an overview of active users and threads, the detailed interactions of different user groups over time, and the dynamic patterns of threads.

Unlike the above-mentioned works, CCVis visualizes both the summarized and detailed clickstream behavior patterns and allows one to drill down for investigating comparative performance and grade implication.

Design Requirements

The design requirements are formed based on multiple sessions of discussion with a campus team of learning scientists, designers, and engineers. The primary goal of developing the CCVis visual analytics framework is to allow instructors to monitor and manage student progress and performance using the course clicksteam data. In particular, it would be ideal if CCVis could help instructors identify, as early as possible, students at risk of failing, so that they can help them adjust or correct their behavior (i.e., the interaction with the online course material) to boost their course performance. Therefore, our interface should fulfill the following design requirements in order to best meet the overall goal.

R1. Provide an overview of the data. The data set used in this research contains clickstreams from 2,000+ students across 14 assignments, and as such, it is impossible to display every clickstream individually. Instead, the visualization should provide an overview of the data that allows users to quickly understand the overall clickstream patterns and the relationships between different types of clicks (e.g., text reading, video watching, etc.).

R2. Visualize student behavior patterns. Instead of drawing connections between individual activities in the clickstream, which fails to capture the complexity of the data, we focus on understanding *behavior patterns*, which are captured by sequences of such activities. For example, a student who previews assignment questions, views study material, and then submits an assignment is likely to perform better than a student who does not preview questions. If we displayed only adjacent click connections, both cases would appear to be identical: a student views study material and then submits the assignment. Thus we make use of behavior patterns to observe how a sequence of clicks, rather than individual clicks, influences future actions. The visualization should allow displaying the connections between different behavior patterns and displaying the actual clicks contained within each behavior pattern.

R3. Group and filter behavior patterns. Although we have 36 unique URLs, they may be categorized into broad types: *reading, video, homepage, and idle times*. Thus two behavior patterns may be classified as having the same *functional role* in our network if the sequence of URL types is the same, allowing us to classify behaviors into *functional groups*, which we hypothesize have different influences on grade distribution. For example, for different assignments, if a student first reads questions and then watches a video, the influence on grade is likely to be the same, even though the unique URLs would be different. The visualization should allow users to easily group or filter behavior patterns according to these groups across different display types, such that users could have a clear understanding of how the groups differ.

R4. Display grade distribution given a behavior pattern. This task is crucial to our goal of determining the probability of a student failing an assignment based solely on the student's clickstream behavior. The visualization should display a behavior pattern together with the corresponding probability distribution of grades that is statistically associated with the behavior pattern, and should also provide a means of comparing the grade distributions of different behavior patterns.

R5. Compare behavior patterns of individual students. In order to identify a student at-risk-of-failing, an instructor might find it useful to compare a student's clickstream with that of another. For example, if a student's clickstream closely resembles that of a student who performed poorly in an assignment, there should be a cause for concern. The visualization should allow users to compare the clickstreams or behavior patterns of two given students across different assignments. Because there are a large number of unique behavior patterns, the visualization should make use of functional role classification stated in **R3**.

R6. Compare behavior patterns across student groups. Having grouped the behaviors into functional groups according to their functional roles (refer to **R3**), we conjecture that different functional groups have different influences on grade distribution. To investigate this hypothesis, the visualization should allow comparing average student functional group distributions for different grade brackets. These comparisons should reveal which functional groups contain behavior patterns that have a strong influence on the final grade.

Data Analysis

In this section, we briefly introduce the main techniques used to analyze the clickstream data: *higher-order networks* (HONs) and *structural identity classification*. HONs extract the critical activity sequences to describe the behavior patterns of students (**R2**). Unlike previous approaches for sequence visualization, which extract the common sequences, we apply HON to identify the *sequences that make a difference to the subsequent activities* and synthesize the *transitions among the sequences*. The graph structure of HON further allows the functional roles of behaviors to be discovered as structural identities (**R3**) using struc2vec, a graph analysis technique.

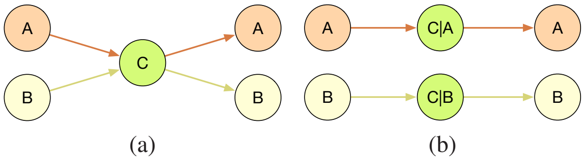
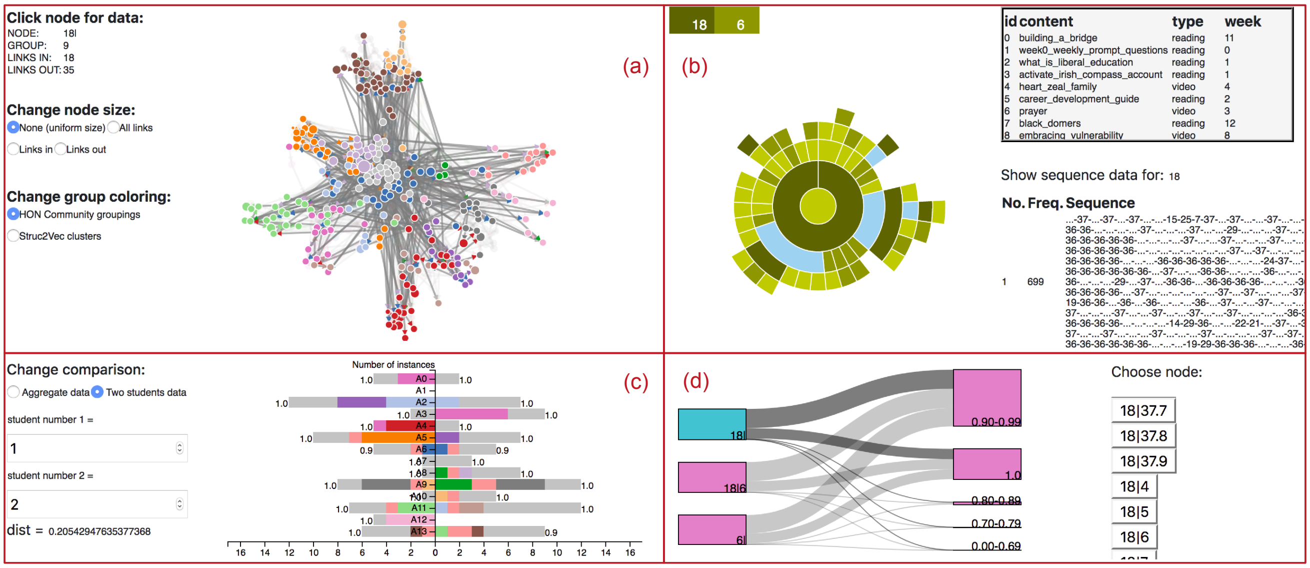


Figure 1. *Comparison between (a) FON and (b) HON representations. A = reading resource A, URL corresponding to assignment 2; B = reading resource B, URL corresponding to assignment 3; C = idle time of 30-45 minutes.*

Higher-order Networks

This clickstream data set has been analyzed previously with a traditional *first-order network* (FON) representation, which does not take into account any higher-order dependency of clicks in the stream. We observe that assigning a single webpage to each node in the network fails to encapsulate the complexity of the clickstream data, as illustrated in the example shown in Figure 1.

In Figure 1 (a), we do not keep track of the history of a sequence; when in an idle time C, the probability of progressing to reading resource A is approximately the same as the probability of progressing to reading resource B. In Figure 1 (b), we rewrite our intermediate node so as to include previous steps in the sequence, creating a HON. Now, we are almost guaranteed to progress from C to A if A was our previous step, and likewise with B. Intuitively this makes sense, as the two reading resources correspond to different assignments. Thus by making use of the recently introduced HON algorithm [29, 22], we can generate an edge table for the clickstream data set that keeps track of previous steps only if they have an influence on future progression.



Click node for data:
NODE: 18l
GROUP: 9
LINKS IN: 18
LINKS OUT: 35

Change node size:
● None (uniform size) ○ All links
○ Links in ○ Links out

Change group coloring:
● HON Community groupings
○ Struc2Vec clusters

id	content	type	week
0	building_a_bridge	reading	11
1	week0_weekly_prompt_questions	reading	0
2	what_is_liberal_education	reading	1
3	activate_irish_compass_account	reading	1
4	heart_zeal_family	video	4
5	career_development_guide	reading	2
6	prayer	video	3
7	black_domers	reading	12
8	embracing_vulnerability	video	8

Show sequence data for: 18

No.	Freq.	Sequence
1	699	...-37-...-37-...-37-...-...-15-25-7-37-...-37-...-...-37-...-...- 36-36-...-...-...-...-37-...-...-...-37-...-...-29-...-...-...-37-...-. 36-36-36-36-36-...-...-...-...-37-...-...-37-...-...-...-37-...-... 36-36-36-36-36-...-...-...-...-...-...-...-37-...-37-...-...-...- 36-36-36-36-...-...-...-36-36-36-36-36-...-...-...-24-37-...- 36-36-36-36-36-36-...-37-...-...-36-36-...-...-...-36-...-...-. 36-...-...-...-29-...-37-...-36-36-...-36-36-36-36-...-...-...-3 36-36-36-36-...-37-...-...-...-...-...-...-37-...-37-...-...-...-37 19-36-36-...-36-...-36-...-...-...-36-...-37-...-...-37-...-...-... 37-...-...-...-37-...-...-37-...-...-37-...-...-37-...-...-...-...-36-3 36-36-36-36-...-...-...-...-14-29-36-...-...-22-21-...-37-...-3 37-...-...-37-...-...-...-37-...-...-...-37-...-...-...-...-37-...-37-. 36-36-36-36-36-36-...-...-...-19-29-36-36-36-...-...-...-36

Change comparison:
○ Aggregate data ● Two students data
student number 1 = 1
student number 2 = 2
dist = 0.20542947635377368

Choose node:
18|37.7
18|37.8
18|37.9
18|4
18|5
18|6

Figure 2. *The screenshot of the CCVis interface. (a) to (d) are the behavior pattern view (BPV), behavior breakdown view (BBV), clickstream comparative view (CCV), and grade distribution view (GDV), respectively. The screenshot shows the interface when the homepage (18) is selected for inspection.*

In this way, we can use HON nodes to represent behavior patterns, assisting us in the design requirement **R2**. The HON algorithm has parameters that can be adjusted such that very rare behavior patterns can be ignored. Furthermore, the number of activities in a given behavior pattern (i.e., the order of the corresponding HON node) is limited by the HON algorithm; the order only increases if moving to a higher order would measurably improve our ability to predict the next node in the sequence. To identify the closely related behaviors, we further apply the widely-used Louvain method [4] to detect communities on HON, using the edge weights and default resolution of 1.0.

Structural Identity Classification

The structural identity of a node can be briefly summarized as a measure of how that node fits into its network: how many direct connections the node has; how many connections its direct connections have; and so on, with precision ultimately limited by what the entire network looks like from that node. struc2vec [19] generates vectors based on this concept of identity, where two nodes having a similar structural identity will be in similar positions in the vector space. The structural similarity of two nodes is derived from their context in the graph, produced by random walks on a corresponding multilayer graph. For example, suppose node *A* acts as a hub node, having a large number of connections with its connections also being of high degree. Also suppose node *B* and node *C* both act as satellite nodes, with just one or two connections with other nodes, though not necessarily with each other. Then the vectors *A* and *B* generated by the struc2vec algorithm will be positioned far apart from each other, whereas vectors *B* and *C* will be positioned close together. Applying the k-means clustering to the generated vectors, therefore, allows us to assign a structural classification to each node, as nodes that are clustered together in the struc2vec space have a similar structural identity. The classification assigned by this technique identifies not only nodes of similar structural identity within the HON, but also behavior patterns of a similar functional role within the clickstreams to meet the design requirement **R3**.

Visual Interface and Interaction

We develop CCVis as a web-based tool for exploring the course clickstream data. The implementation uses D3.js for producing dynamic and interactive data visualizations in web browsers. As shown in Figure 2, our CCVis consists of four components: the behavior pattern view (BPV), behavior breakdown view (BBV), clickstream comparative view (CCV), and grade distribution view (GDV). The BPV gives users an overview of the entire data set as a graph (i.e., HON) where nodes denote sequences of click tracks (higher-order nodes) and edges show their interconnections. The BBV allows one to check each individual higher-order node in a coarse-to-fine manner using the sunburst diagram. The CCV shows detailed comparison of two students' clickstream data on a weekly basis as well as the average clickstream content of all grade brackets. Finally, the GDV displays using the Sankey diagram, the mapping between sequences of click tracks and grade brackets based on statistics. All these views are dynamically linked together via standard brushing and linking. In the following, we describe each view in detail.

Behavior Pattern View (BPV)

The BPV corresponds to the design requirements **R1** and **R2**. By incorporating the concept of HONs into this view, we are able to summarize a large number of clicks into a manageable set of nodes and edges for effective viewing and exploration. For a clear overview of the clickstream data, we apply the force-directed graph layout to draw the BPV where each node in the graph represents a higher-order node (i.e., a behavior consisting of a sequence of clicks) and each edge represents the connection between the two incident nodes. As shown in Figure 2 (a), we map the strength of connection to edge opacity (dark for high strength and gray for low strength). Users can choose to map node size to the number of links both entering and exiting the node. Nodes are grouped via struc2vec and node color denotes group membership. These visual mappings allow users to quickly find useful information (e.g., spotting important nodes or identifying group distribution) for further exploration. By displaying how different behavior patterns interact, we are able to identify strong patterns of behavior and draw conclusions on the role that each behavior pattern plays on student learning.

Behavior Breakdown View (BBV)

While the BPV provides a clear overview of the learning behavior patterns, it fails to explain the actual click tracks within these identified behaviors. For a complete investigation, a more thorough understanding of these behavior patterns is required. Therefore, we design the BBV that utilizes the sunburst diagram to obtain the additional required insight into what was happening within each behavior pattern. The BBV corresponds to the design requirement **R3**. It details the behavior data as the sequences of clicks they represent, allowing users to view in detail specific nodes within the BPV and categorize them into groups based on which point in the sequence an activity occurs.

As shown in Figure 2 (b), users can find each behavior pattern displayed as a sequence of colored sections. By reading these sequences from the center outward, they can carefully examine any node in the BPV. The size of each section indicates the percentage of students who followed that specific sequence and each different color represents a specific type of URL click. As we proceed outward, an active labeling system is shown on top indicating which sequence has been selected. However, as the sequences progress outward, the sections could become too small for normal viewing. As such, we leverage the zoomable version of the sunburst diagram so that users can zoom in the diagram.

The BBV clarifies the details within behavior patterns found in the BPV. From this, users are able to determine the frequency of sequences by observing the size of each section and see how a sequence of clicks leads to a certain behavior. Users can also see how a sequence of clicks can branch into multiple behaviors as they follow different click paths. The BPV shares a direct connection with the BBV, in which the BBV can draw more specific conclusions from the BPV and vice versa. Through brushing and linking, when a node in the BPV is selected, the corresponding sequence in the BBV is also selected. When users select a sequence from the BBV, multiple nodes may be selected. If users choose to select a more general pattern (e.g., a first-order pattern), all nodes that branch from this pattern will be selected, making it easier to see the role of the pattern in the BPV.

Clickstream Comparative View (CCV)

Corresponding to the design requirements **R5** and **R6**, the CCV allows users to observe a student's clickstream data throughout the course, as well as an aggregated breakdown of how student behaviors correlate with grades. Furthermore, we wish to categorize behavior patterns according to functional group in order to simplify the display, rather than cluttering up the display with a record of individual behavior patterns. We make use of the stacked bar chart layout to display this categorized data.

In the "Two students data" option of the CCV, we use a mirrored stacked bar chart to compare the clickstream content of two students. The two students are selected by their IDs, as shown in Figure 2 (c). The frequency of functional group members is defined as the number of times a behavior pattern from that functional group is identified in the indicated clickstream and is displayed on the *x*-axis; week number is indicated on the *y*-axis. In this view, the goal is to compare clickstream content between two students for any given week. In the "Aggregate data" option of the CCV, we use a non-mirrored stacked bar chart to compare the average clickstream content of all grade brackets. Since the view is no longer mirrored, we find it clearer to display the average frequency of functional group members on the *y*-axis and the grade bracket on the *x*-axis. Figure 10 shows such an example.

The CCV allows us to compare behavior patterns of different students in a search for either similarity (in the case of two students scoring the same grade) or differences (where two students score differently). Furthermore, it allows an instructor to see at a glance how much work a student is putting in when grading an assignment: if an instructor notices an unusually low frequency of certain functional groups, it should serve as a warning to examine the student's work more closely. In the BPV, users may choose to color the nodes according to functional group, which are the same as those displayed in the CCV. This allows users to determine meaningful relationships between the nature of a functional group and its influence on grade.

Grade Distribution View (GDV)

While the BPV and BBV allow users to observe the behavior patterns themselves and how they interact with one another, at this point we still have not tied behavior patterns found in the clickstream data to grade distributions. The challenge then is to tie a behavior to a grade distribution. Given the information from the BBV, BPV, and CCV, we can now objectively compare students. This is achieved by using the Sankey diagram to correlate URL sequences (i.e., behavior patterns) to grade distributions. As shown in Figure 2 (d), the GDV allows users to see the informed grade distributions of each sequence based on the data provided, corresponding to the design requirement **R4**. Users are able to quickly observe which grades are (not) likely to occur given a behavior pattern. In addition, the GDV can potentially be used to make grade predictions for future students based on the clickstream data of past students.

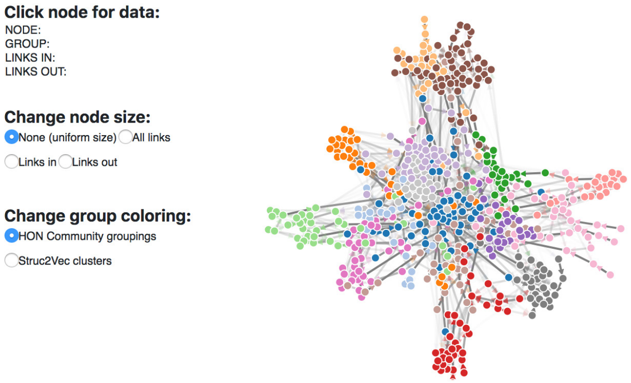
Results and Discussion

Our CCVis is released online at: `http://www.nd.edu/~cwang11/ccvis/`. To avoid any compatibility issues (known problems include deleting a higher-order node from the GDV), we recommend users to use the Mozilla Firefox browser. In the following, we present five case studies and highlight the insights gleaned. The five studies jointly cover all six design requirements. Then, we report the evaluation given by a group of experts including learning scientists, designers, and engineers.

Case Studies

Case Study 1: Overview of behavior data. This case study demonstrates users gaining an overview of the data using the BPV in order to develop an intuitive understanding of both HON community groupings and struc2vec clusters. The design requirements **R1** and **R2** are covered here.

Users begin with the default view for the BPV, where they observe clear community groupings within the network as shown

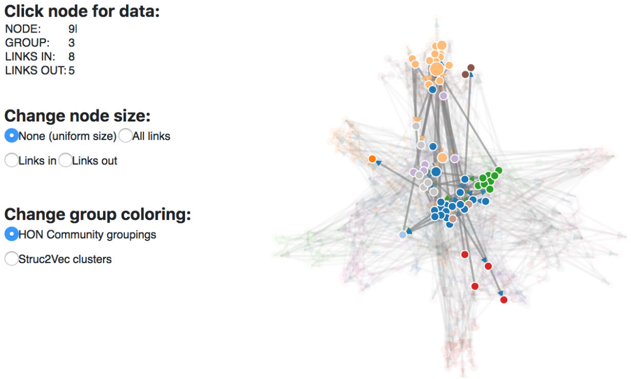


Click node for data:
NODE:
GROUP:
LINKS IN:
LINKS OUT:

Change node size:
● None (uniform size) ○ All links
○ Links in ○ Links out

Change group coloring:
● HON Community groupings
○ Struc2Vec clusters

__Figure 3.__ The default view for the BPV showing HON community groupings.



Click node for data:
NODE: 9|
GROUP: 3
LINKS IN: 8
LINKS OUT: 5

Change node size:
● None (uniform size) ○ All links
○ Links in ○ Links out

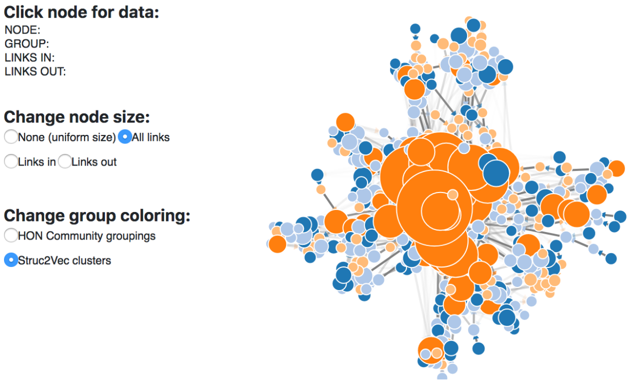
Change group coloring:
● HON Community groupings
○ Struc2Vec clusters

__Figure 4.__ The highlight view for the BPV where the higher-order node 9| is selected.

in Figure 3. Users then select a higher-order node as shown in Figure 4, allowing them to focus on the details of that particular node. The size of the selected node is increased, and any nodes in the network that are not directly connected to the node of interest become transparent.

Furthermore, the node data displayed at the top-left corner can be used to determine that the community groupings roughly correspond to clicks made during individual weeks in the semester. For example, the highlighted node 9| is an activity belonging to week 7, as can be determined by using the activity table shown at the top-right corner of Figure 2 (b). Users may then choose to click one of the nodes connected to this selected node in order to observe the activities present in that node, following one possible path for a student's clickstream to better understand the local neighborhood. In this case, it is found that the majority of nodes contained in group 3 contain activities belonging to week 7, with one node, 9|18.37.29, illustrating a "boundary node" in that it contains both an activity belonging to week 7 (id: 9) and an activity belonging to week 6 (id: 29).

By double-clicking outside of the selected node users return to the default view, and change the group coloring according to struc2vec clustering. Since the struc2vec clusters are determined according to structural identity, which has a close relationship to node degree, users find it useful to change node size so as to indicate the number of connections each node has, as shown in Figure 5. By hovering over each node, users can quickly develop a coherent idea of what type of nodes are contained in each structural classification.



Click node for data:
NODE:
GROUP:
LINKS IN:
LINKS OUT:

Change node size:
○ None (uniform size) ● All links
○ Links in ○ Links out

Change group coloring:
○ HON Community groupings
● Struc2Vec clusters

__Figure 5.__ BPV color-coded according to struc2vec clusters, with node size proportional to the number of connections.

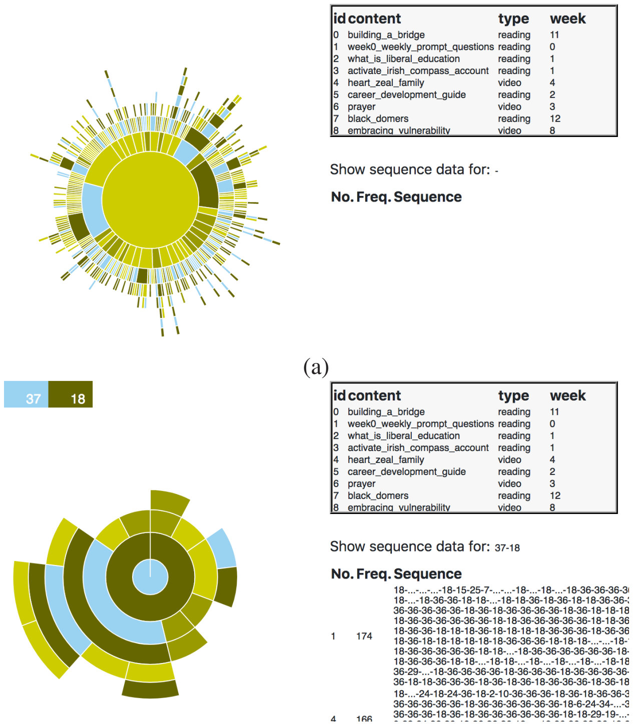
Users first observe that the larger, orange nodes correspond to 'hub'-type nodes. When located centrally, they act as global hubs: higher-order nodes consisting of various combinations of idle time and the course homepage, e.g., 37|18.36, 18|36. These are nodes that would be found throughout the duration of the semester and are thus important components for most of the clickstreams. When located away from the center, they act as local hubs: typically either a single activity such as 5|, or an activity following the course homepage, such as 7|18. The dark blue nodes appear very similar to the orange nodes, though typically with fewer connections than the orange nodes.

The pale blue grouping has a high proportion of nodes that contain two activities from the same week, e.g., 0|20 from week 11, 12|27 from week 9, and 33|21 from week 4, etc. The pale orange grouping also has several nodes with two or even three activities from the same week, but users note that this classification also contains many "boundary nodes". Examples include the aforementioned 9|18.37.29 with week 7 (id: 9) and week 6 (id: 29). As a result of containing nodes with activities spanning across assignments, the pale orange grouping nodes typically have very small degrees and occur rarely in student clickstreams.

Case Study 2: Detailed view of behavior patterns. Now that users have a thorough grasp of the relationship between nodes, we move to a case study in which users aim to investigate the detailed click pathways of the behavior patterns using the BBV. The design requirement **R3** is covered here.

Users begin with the default view, as shown in Figure 6 (a), where they can see at a glance what the content of each behavior pattern's click pathway is—light yellow URLs correspond to *reading*, medium yellow to *video*, dark yellow to the *homepage*, and light blue to *idle times*. Users click on a given sequence to zoom in on the rest of that sequence's click path, as shown in Figure 6 (b) where the sequence 18|37 is selected for inspection. As shown at the bottom-right corner, the sequence data table lists every student whose clickstream contains the highlighted sequence, together with the frequency of that sequence within their clickstream and a copy of their clickstream. This helps users identify how common the sequence is and in what context it generally appears.

Case Study 3: Determining grade distribution for a given clickstream. This case study features users investigating what



(a)

id	content	type	week
0	building_a_bridge	reading	11
1	week0_weekly_prompt_questions	reading	0
2	what_is_liberal_education	reading	1
3	activate_irish_compass_account	reading	1
4	heart_zeal_family	video	4
5	career_development_guide	reading	2
6	prayer	video	3
7	black_domers	reading	12
8	embracing_vulnerability	video	8

Show sequence data for: -

No. Freq. Sequence

(b)

id	content	type	week
0	building_a_bridge	reading	11
1	week0_weekly_prompt_questions	reading	0
2	what_is_liberal_education	reading	1
3	activate_irish_compass_account	reading	1
4	heart_zeal_family	video	4
5	career_development_guide	reading	2
6	prayer	video	3
7	black_domers	reading	12
8	embracing_vulnerability	video	8

Show sequence data for: 37-18

No.	Freq.	Sequence
1	174	18-...-...-...-18-15-25-7-...-...-18-...-18-...-18-36-36-36-3 18-...-18-36-36-18-18-...-18-18-36-18-36-18-18-36-36- 36-36-36-36-36-18-36-18-36-36-36-36-18-36-18-18-18 18-36-36-36-36-36-36-18-18-36-36-36-36-36-18-18-36 18-36-36-18-18-18-36-18-18-18-18-36-36-36-18-36-36 18-36-18-18-18-18-18-18-36-36-36-18-18-18-...-...-18- 18-36-36-36-36-36-18-18-...-18-36-36-36-36-36-36-18- 18-36-36-36-18-18-...-18-18-...-18-...-18-...-18-...-18-18 36-29-...-18-36-36-36-36-18-36-36-36-18-36-36-36-36- 36-18-18-36-36-36-18-36-18-36-18-36-36-36-18-36-18
4	166	18-...-24-18-24-36-18-2-10-36-36-36-18-36-18-36-36-3 36-36-36-36-36-18-36-36-36-36-36-36-18-6-24-34-...-3 36-36-36-18-36-18-36-36-36-36-36-36-18-18-29-19-...-

Figure 6. *Comparison between (a) default BBV and (b) BBV when the sequence* 18|37 *is selected for inspection.*

grade distributions certain behavior patterns are associated with, leveraging the GDV and fulfilling the design requirement **R4**.

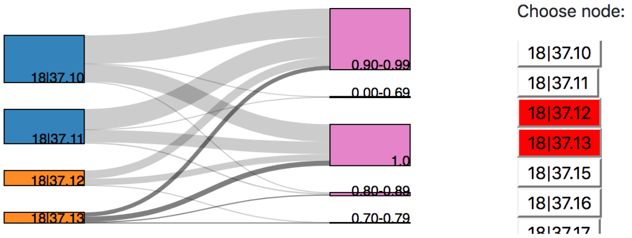
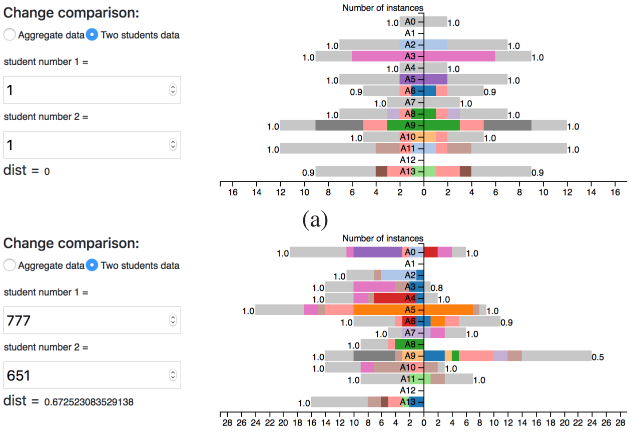


Figure 7. *Grade distributions for two typical nodes* (18|37.10 *and* 18|37.11) *and two outlier nodes* (18|37.12 *and* 18|37.13) *in the GDV. Node colors are based on their* struc2vec *cluster colors shown in Figure 5.*

Users may select up to four behavior patterns at a time in order to compare how they influence grades. The node selection box, as shown on the right in Figure 7, colors nodes that give a 'typical' grade distribution white, whereas 'outlier' nodes are colored in red. Selecting two outliers to compare with two typical nodes does not yield much insight. Due to the fact that this data set has very little variation in the results received, our 'outliers' are not very different from the typical nodes. If there were more variance in the results, users might leverage the GDV to identify nodes of interest to study further using the BBV and BPV.

Case Study 4: Comparing clickstream content of individual students. In this case study, users wish to better understand why some students perform better than others by comparing clickstream content of individual students throughout the semester. Users make use of the CCV and BPV to compare individual student clickstreams and to explain the discrepancy in grade for a given assignment for two students. In addition, users make use of the BBV to identify students sharing a common feature. This case study covers the design requirement **R5**.


Change comparison:

○ Aggregate data ● Two students data

student number 1 = 1

student number 2 = 1

dist = 0

(a)

Change comparison:

○ Aggregate data ● Two students data

student number 1 = 777

student number 2 = 651

dist = 0.672523083529138

(b)

Figure 8. *Comparison between (a) the default view for CCV and (b) CCV after "student number 1" is changed manually, automatically updating "student number 2". Colors shown in the stacked bar chart are based on HON community group colors shown in Figure 3.*

Users begin with the default view for the CCV, as shown in Figure 8 (a), and immediately seek to explain why, if the communities are typically associated with specific weeks, the light gray community appears to dominate every assignment. By analyzing the BPV in the same way as we did in Figure 4 for Case Study 1, users determine that the gray grouping consists largely of activities from week 10, with a single node of 18|36 (homepage | idle time, a sequence found consistently throughout all clickstreams) causing the community to be over-represented in all assignments.

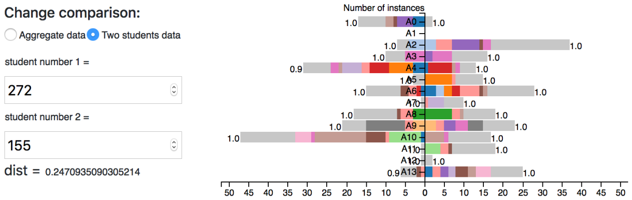
Satisfied that the light gray community has been explained, users wish to compare students with dissimilar clickstream contents, as one would expect them to have different grades. When "student number 1" is changed by users, "student number 2" is automatically updated to the student with the greatest distance from "student number 1" (indicated at the bottom-left corner) i.e., the student whose clickstream content is most different, as shown in Figure 8 (b). Users note that the student on the right in Figure 8 (b), student 777, performed poorly in assignment A9, scoring only 50%. In contrast, student 651 scored 100%.

Users observe that student 777 has significantly more clicks in assignment A9 than student 651, and would, therefore, be expected to perform better. Users note that the only communities in which student 777 has fewer clicks than student 651 are the dark gray community and the pale orange community. Users learn through analyzing the BPV in Case Study 1 that the pale orange community is associated with week 7, and by analyzing the BPV again we find that the dark gray community is most strongly associated with week 4. Thus users conclude that student 777's poor performance is likely influenced most by his/her relative lack of behavior patterns corresponding to the pale orange community, i.e., a lack of clicks corresponding to that week's assignment.

Finally, users would like to compare how two students with

View	Description	Requirements	User & Practical Application Question
BPV	HON	**R1**, **R2**	Data Scientist, Learning Designer: What are the strongest relationships of click behavior patterns?
BBV	Sunburst	**R3**	Data Scientist, Learning Designer: What are the most popular students clickstream pathways for accessing and engaging with the course?
CCV	Stacked Bar Chart	**R5**, **R6**	Instructor: Did my students click on any videos or articles before turning in the homework?
GDV	Sankey Diagram	**R4**	Data Scientist: How can I reverse-engineer and figure out the students who got the lowest grade so I can describe and predict future student behavior? Program Director: What behavior pattern correlates to a grade distribution?

The summary of the four views of CCVis, their description, requirements, and user and practical application question.



Change comparison:

○ Aggregate data ● Two students data

student number 1 = 272

student number 2 = 155

dist = 0.2470935090305214

Figure 9. *Two students with the same frequency of the sequence* $9|18.36$, *associated with assignment A9.*

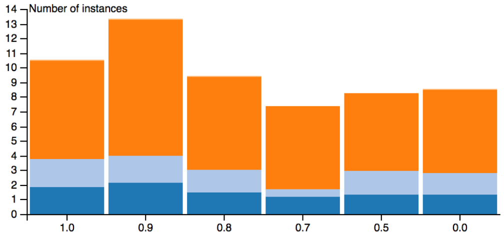


Figure 10. *Student clickstream data averaged over each grade bracket in the CCV. Colors shown in the stacked bar chart are based on their* struc2vec *cluster colors shown in Figure 5. Note that the pale orange band is too small to be visible as nodes in this group occur quite rarely.*

similar frequencies of a given sequence compare with one another, in order to investigate two students with similar clickstream content rather than very different clickstream content. Users select the node $9|18.36$ in the pale orange community of the BPV, knowing that it corresponds to assignment A9. The sequence $9|18.36$ is then highlighted in the BBV as well, similar to Figure 6 (b). The sequence data table accompanying the zoomed BBV indicates that students 272 and 155 jointly have the highest frequency of this node, with two occurrences in each student's clickstream. Users then manually set "student number 1" to be 272 and "student number 2" to be 155, as shown in Figure 9. It is immediately apparent to users that both students have an almost identical proportion of pale orange community content—the community most strongly associated with week 7 (the week of assignment 9) —and that both students have the same grade. In this way, users are able to identify high-frequency behavior patterns from any community, observe general trends, and investigate how they influence a student's grade.

Case Study 5: Comparing clickstream content of students grouped by grade. In this case study, users wish to use the average clickstream content of students across all assignments (separated according to grade received) to determine which of the structural classifications described in Case Study 1 might best serve as a proxy for grade prediction. This case study covers the design requirement **R6**, and makes use of both the CCV and BPV.

Users change group coloring to struc2vec clusters, and change the CCV comparison to "Aggregate data", as shown in Figure 10. The coloring in the aggregated view is the same as that used in Figure 5, though users note the absence of the pale orange grouping. As the pale orange grouping occurs very infrequently throughout student clickstreams, the average number of pale orange nodes is very small, and is not discernible in the aggregated view. Users note that students who scored in the range of 90-99% have the largest number of clicks, which at first glance seems unusual. However, students who are very confident in the material for their course are less likely to review it before completing assignments, and would get high grades anyway. A student who has not gotten a firm grasp of the material to begin with is likely to spend far more time familiarizing themselves with the content, i.e., producing more clicks.

Close inspection of Figure 10 indicates that, excluding grade 1.0, the strongest indicator of student grade is obtained by looking at the frequency of dark blue grouping behavior patterns, followed closely by the orange grouping behavior patterns. Recalling the discussion in Case Study 1 about the nature of these communities, users note that these groupings correspond to hubs, typically consisting of either a single activity (e.g., $5|$), an activity following the homepage (e.g., $7|18$), or one of a number of combinations of idle time and course homepage (e.g., $37|18.36$, $18|36$). It makes sense that nodes which indicate a student beginning work on an assignment—homepage to single activity—would be strongly correlated with grade. Furthermore, a large number of idle time nodes could be indicative of a student taking his/her time to read through course material thoroughly, as short idle times correspond to just 15 minutes of inactivity.

Expert Evaluation

In the above table, a summary of the four views of CCVis is provided with a description of the visual representation, their associated requirements, and practical application questions for end users. A detailed analysis of each view's benefits and limitations from a campus team of learning scientists, designers, and engineers follows the summary.

Behavior Pattern View (BPV): The overall clickstream patterns revealed by the BPV can help users quickly spot the most important patterns and identify the relationships between different patterns. Users can also further investigate the patterns by clicking on an individual node to discover which group the node belongs to and how many entering and exiting links it has. The option of mapping node size to its total link numbers and entering or exiting link quantity helps users easily identify the popularity of a node. Furthermore, the "HON Community groupings" option clearly shows the activity patterns in each individual week throughout the semester. We found the option of grouping patterns by "struc2vec clusters" less intuitive and the meaning of each color is less clear. The inclusion of a color legend would help users better interpret the graph.

Behavior Breakdown View (BBV): The sunburst diagram clarifies the details within the pattern identified in the BPV so users can dive deeper and discover the detailed click pathways of any interesting patterns. Users can also determine the frequency of a specific pathway by the size of the section. Additionally, the direct connection between the BBV and BPV enables users to investigate how a click pathway leads to a certain behavior and what role it plays in the overall pattern. We found it a challenge to determine the exact click types when brushing outward because the labeling system only displays the coded number. Although there is a table legend at the top-right corner for lookup, users have to use the scroll bar to find the click type for the code they are investigating.

Clickstream Comparative View (CCV): The mirrored stacked bar chart clearly shows the difference between student activity patterns in each week, which reveals their different learning interests, styles, and strategies. It also uncovers the weeks where students did not have any course activities, which could be an indicator that a student might be experiencing some academic or personal challenges. Proper and timely action taken by instructors or advisors can help the student overcome the challenges and thrive. The grades scale (0-1) displayed in the "Aggregate data" view does not match with the practiced scale (0-20), requiring an extra step as users comprehend the chart.

Grade Distribution View (GDV): The Sankey diagram reveals the correlation between behavior patterns and grade distributions. The visual impact is very effective. Users can select an interesting behavior pattern and investigate what grade distribution it leads to. The view also enables users to discover what grades may (not) occur given a certain behavior pattern. Again, we found it a challenge to determine the exact activity type by the coded numbers. Also, it would help users interpret the chart more efficiently if the grade distribution on the right could be displayed in a fixed order while investigating different nodes.

In summary, CCVis is an engaging tool that can help data scientists, learning designers, and program directors discover the overall patterns of student learning activities, the detailed pathways of their clickstreams, and the correlation between an activity pattern and the grade distribution. It is a critical foundational step in our effort to comprehensively analyze student learning behaviors, draw actionable insights from the analysis, and take proper actions based on these insights to help students thrive.

To further develop CCVis into a tool that can benefit instructors, advisors, and students as well, we have the following suggestions: (1) Develop an aggregated view of the entire class clickstream pattern so instructors or advisors could use it as a benchmark when investigating a specific students' week-by-week click behavior. Students could also benefit from this by seeing where they stand in terms of course activity compared with their peers. (2) Connect all the four views so users could interact with them live together. For example, when users click on a node in the BPV, the detailed pathway within that node would be highlighted in the BBV. Meanwhile, the CCV could compare the overall activity pattern of the student who demonstrated that behavior pattern the most with his/her aggregated class. Concurrently, the GDV could display the grade distribution to which the pattern leads. (3) Provide a query option for instructors or advisors to quickly identify students and their clickstream patterns. Permission control should be implemented so instructors or advisors could only query the students who are in their class or are their advisees. (4) Provide a query for users to discover the behavior patterns of students who have lower grades.

Limitations

We observe the following limitations of the current work. First, although the course clickstream data we collected from the FYE course include 2.3 million of click tracks from 2,000+ students, the grade distribution was predominantly in the A range. Since the grades are highly skewed toward one extreme, it became very difficult for us to make an accurate prediction of the final grade using the clickstream behavior patterns. We actually attempted the use of deep neural networks to make predictions but the results are not good as the data records are highly unbalanced. This, however, would not become a problem as our learning engineers would collect similar data for more challenging introductory courses such as Introduction to Chemistry. We expect the grade distribution to be more balanced, increasing the possibility of more accurate student performance prediction using clickstream behavior patterns. Second, the current version of CCVis focuses on behavior patterns described by the activity sequences corresponding to the higher-order nodes. However, it is possible that more complicated patterns should be analyzed using larger-scale features in the HON, e.g., paths and motifs. These features in the HON will provide more contextual information of how a sequence of activities associated with a higher-order node is performed. For example, a path in the HON can describe the entire activity sequence for a student to complete one assignment. Displaying paths on the BPV, instructors can immediately perceive all activities performed by one student or visually compare multiple students to understand their behavioral differences.

Conclusions and Future Work

In this work, we design, demonstrate, and evaluate CCVis, a visual analytics tool for analyzing student course clickstream data and exploring student online learning behaviors. In the future, we would improve the visual encodings of CCVis. For example, the node-link diagram shown in the BPV could easily suffer from the scalability and occlusion issues. The sunburst diagram shown in the BBV is aesthetically pleasing and space-filling but could be difficult to read. All these issues should be addressed. CCVis is currently designed for instructors to monitor and manage student progress and performance. We could go beyond the clickstream data and collect student writing and instructor feedback for text mining. This could help instructors facilitate their grading of writing assignments and better spot students at risk. Besides instructors, we could also design and customize different versions of this analytics tool to serve individual students and administrators. Students could benefit from such a tool by recognizing their behaviors and performance comparing to peers, potentially promoting self-motivation in their study. The administrators (e.g., the program director, dean, and provost) could benefit from such a tool by gaining an informative overview to accurately understand or estimate student retention.

Acknowledgments

This work was supported in part by the U.S. National Science Foundation through grants IIS-1455886, IIS-1560363, and DUE-1833129.

References

[1] K. E. Arnold and M. D. Pistilli. Course signals at Purdue: Using learning analytics to increase student success. In *Proceedings of International Conference on Learning Analytics and Knowledge*, pages 267–270, 2012.

[2] B. O. Barefoot, J. N. Gardner, M. Cutright, L. V. Morris, C. C. Schroeder, S. W. Schwartz, M. J. Siegel, and R. L. Swing. *Achieving and Sustaining Institutional Excellence for the First Year of College*. John Wiley & Sons, 2010.

[3] J. P. Bean. Increasing student retention: Effective programs and practices for reducing the dropout rate, 1987.

[4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[5] P. Bodesinsky, B. Alsallakh, T. Gschwandtner, and S. Miksch. Exploration and assessment of event data. In *Proceedings of EuroVis Workshop on Visual Analytics*, 2015.

[6] B. C. M. Cappers and J. J. van Wijk. Exploring multivariate event sequences using rules, aggregations, and selections. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):532–541, 2018.

[7] S. Charleer, A. V. Moere, J. Klerkx, K. Verbert, and T. D. Laet. Learning analytics dashboards to support adviser-student dialogue. *IEEE Transactions on Learning Technologies*, 2018.

[8] Q. Chen, Y. Chen, D. Liu, C. Shi, Y. Wu, and H. Qu. PeakVizor: Visual analytics of peaks in video clickstreams from massive open online courses. *IEEE Transactions on Visualization and Computer Graphics*, 22(10):2315–2330, 2016.

[9] Y. Chen, P. Xu, and L. Ren. Sequence synopsis: Optimize visual summary of temporal event data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):45–55, 2018.

[10] L. Derick, G. Sedrakyan, P. J. Munoz-Merino, C. D. Kloos, and K. Verbert. Evaluating emotion visualizations using affectvis, an affect-aware dashboard for students. *Journal of Research in Innovative Teaching & Learning*, 10(2):107–125, 2017.

[11] F. Dernoncourt, C. Taylor, U.-M. O'Reilly, K. Veeramachaneni, S. Wu, C. Do, and S. Halawa. MoocViz: A large scale, open access, collaborative, data analytics platform for moocs. In *NIPS workshop on Data-driven Education*, 2013.

[12] F. Du, B. Shneiderman, C. Plaisant, S. Malik, and A. Perer. Coping with volume and variety in temporal event sequences: Strategies for sharpening analytic focus. *IEEE Transactions on Visualization and Computer Graphics*, 23(6):1636–1649, 2017.

[13] S. Fu, J. Zhao, W. Cui, and H. Qu. Visual analysis of MOOC forums with iForum. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):201–210, 2017.

[14] T.-C. Hsieh and T.-I. Wang. A mining-based approach on discovering courses pattern for constructing suitable learning path. *Expert Systems with Applications*, 37(6):4156–4167, 2010.

[15] Z. Liu, B. Kerr, M. Dontcheva, J. Grover, M. Hoffman, and A. Wilson. CoreFlow: Extracting and visualizing branching patterns from event sequences. *Computer Graphics Forum*, 36(3):527–538, 2017.

[16] S. Malik, B. Shneiderman, F. Du, C. Plaisant, and M. Bjarnadottir. High-volume hypothesis testing: Systematic exploration of event sequence comparisons. *ACM Transactions on Interactive Intelligent Systems*, 6(1):9:1–9:23, 2016.

[17] National Center for Education Statistics. Undergraduate retention and graduation rates. `https://nces.ed.gov/programs/coe/indicator_ctr.asp`.

[18] C. Partl, S. Gratzl, M. Streit, A. M. Wassermann, H. Pfister, D. Schmalstieg, and A. Lex. Pathfinder: Visual analysis of paths in graphs. *Computer Graphics Forum*, 35(3):71–80, 2016.

[19] L. F. R. Ribeiro, P. H. P. Saverese, and D. R. Figueiredo. struc2vec: Learning node representations from structural identity. In *Proceedings of ACM SIGKDD Conference*, pages 385–394, 2017.

[20] C. Shi, S.Fu, Q. Chen, and H. Qu. VisMOOC: Visualizing video clickstream data from massive open online courses. In *Proceedings of IEEE Pacific Visualization Symposium*, pages 159–166, 2015.

[21] M. Steptoe, R. Krüger, R. Garcia, X. Liang, and R. Maciejewski. A visual analytics framework for exploring theme park dynamics. *ACM Transactions on Interactive Intelligent Systems*, 8(1):4:1–4:27, 2018.

[22] J. Tao, J. Xu, C. Wang, and N. V. Chawla. HoNVis: Visualizing and exploring higher-order networks. In *Proceedings of IEEE Pacific Visualization Symposium*, pages 1–10, 2017.

[23] D. Trimm, P. Rheingans, and M. desJardins. Visualizing student histories using clustering and composition. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2809–2818, 2012.

[24] K. Wongsuphasawat, J. A. G. Gómez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. LifeFlow: Visualizing an overview of event sequences. In *Proceedings of ACM SIGCHI Conference*, pages 1747–1756, 2011.

[25] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2659–2668, 2012.

[26] K. Wongsuphasawat, C. Plaisant, M. Taieb-Maimon, and B. Shneiderman. Querying event sequences by exact match or similarity search: Design and empirical evaluation. *Interacting with Computers*, 24(2):55–68, 2012.

[27] D. Wortman and P. Rheingans. Visualizing trends in student performance across computer science courses. *ACM SIGCSE Bulletin*, 39(1):430–434, 2007.

[28] T. Wu, Y. Yao, Y. Duan, X. Fan, and H. Qu. NetworkSeer: Visual analysis for social network in MOOCs. In *Proceed-*

ings of IEEE Pacific Visualization Symposium, pages 194–198, 2016.

[29] J. Xu, T. L. Wickramarathne, and N. V. Chawla. Representing higher-order dependencies in networks. *Science Advances*, 2(5), 2016.

Author Biography

Maggie Celeste Goulden *is an undergraduate student of astrophysics at Trinity College Dublin. She conducted this work under the support of the Naughton Fellowship for REU at University of Notre Dame during Summer 2018.*

Eric Gronda *is an undergraduate student of computer engineering at University of Maryland, Baltimore County. He conducted this work under the support of the NSF DISC (Data Intensive Scientific Computing) REU at University of Notre Dame during Summer 2018.*

Yurou Yang *is an undergraduate student of computer science and technology at Zhejiang University. She conducted this work as an iSURE (International Summer Undergraduate Research Experience) student at University of Notre Dame during Summer 2018.*

Zihang Zhang *is an undergraduate student of computer science and technology at Zhejiang University. He conducted this work as an iSURE (International Summer Undergraduate Research Experience) student at University of Notre Dame during Summer 2018.*

Jun Tao *is a postdoctoral researcher at University of Notre Dame and will join Sun Yat-sen University as an associate professor. He received a Ph.D. degree in computer science from Michigan Technological University in 2015. Dr. Tao's main research interests are scientific visualization and visual analytics.*

Chaoli Wang *is an associate professor of computer science and engineering at University of Notre Dame. He received a Ph.D. degree in computer and information science from The Ohio State University in 2006. Dr. Wang's main research interests are scientific visualization and visual analytics.*

Xiaojing Duan *plays the learning analytics architect role in the Office of Information Technologies, University of Notre Dame. Her primary focuses include building the Learning Record Warehouse, performing data analysis, and developing visualization dashboards in an effort to improve student learning outcomes.*

G. Alex Ambrose *is a digital learning research scientist serving as the associate director of ePortfolio Assessment in Kaneb Center for Teaching and Learning, University of Notre Dame. He received a Ph.D. degree in computing technology in education from Nova Southeastern University in 2013.*

Kevin Abbott *is an educational technology specialist in the Academic Technologies Services team at Office of Information Technologies, University of Notre Dame. He works closely with faculty to choose and implement technology to enhance teaching and learning.*

Patrick Miller *is the lead learning management professional in the Academic Technologies Services team at Office of Information Technologies, University of Notre Dame.*