

EventAction: Visual Analytics for Temporal Event Sequence Recommendation

Fan Du

Catherine Plaisant

Neil Spring

Ben Shneiderman

University of Maryland*

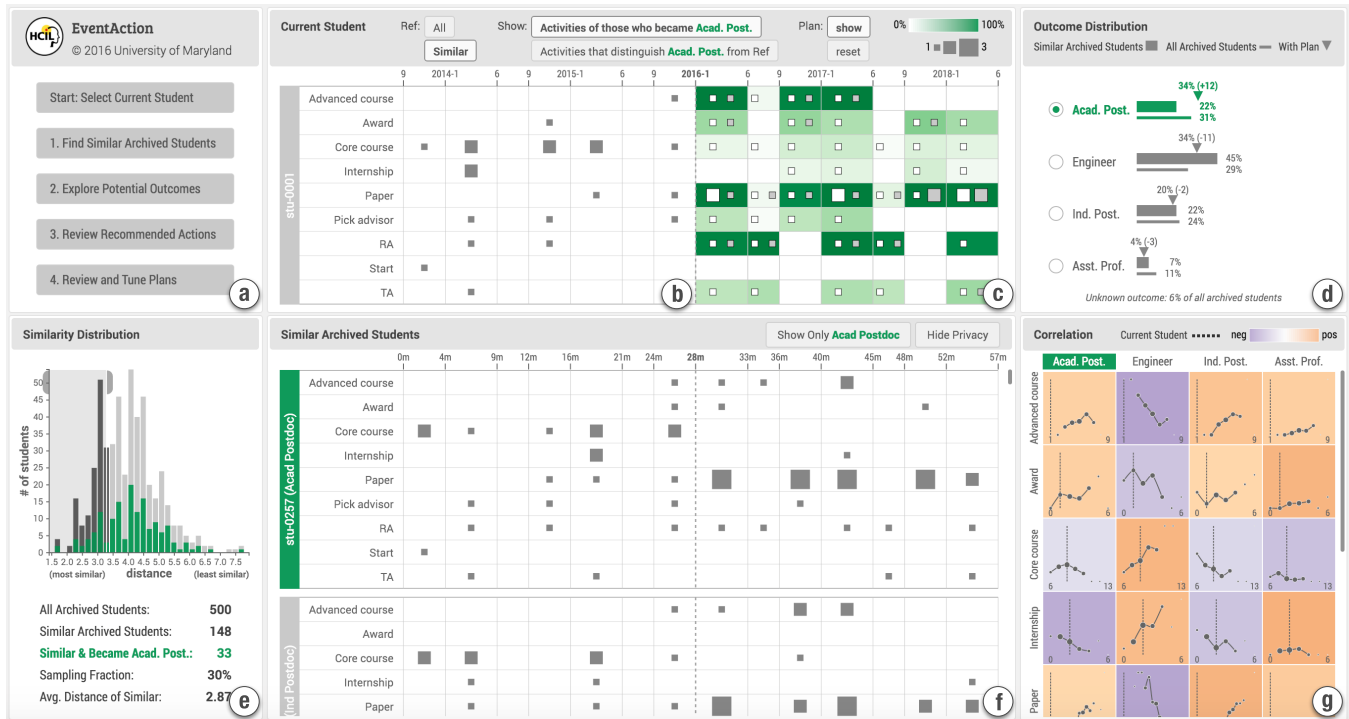


Figure 1: EventAction provides a visual analytics approach for helping data analysts recommend actions to improve the outcome. The user interface consists of seven coordinated views, opening progressively as the analysis progresses: (a) workflow control panel, (b) current record timeline, (c) activity summary view, (d) outcome distribution view, (e) similarity distribution view, (f) similar archived record timelines, and (g) correlation view. Figures in this paper illustrate a synthetic dataset.

ABSTRACT

Recommender systems are being widely used to assist people in making decisions, for example, recommending films to watch or books to buy. Despite its ubiquity, the problem of presenting the recommendations of temporal event sequences has not been studied. We propose EventAction, which to our knowledge, is the first attempt at a prescriptive analytics interface designed to present and explain recommendations of temporal event sequences. EventAction provides a visual analytics approach to (1) identify similar records, (2) explore potential outcomes, (3) review recommended temporal event sequences that might help achieve the users' goals, and (4) interactively assist users as they define a personalized action plan associated with a probability of success. Following the design study framework, we designed and deployed EventAction in the context of student advising and reported on the evaluation

*e-mails: {fan,plaisant,nspring,ben}@cs.umd.edu

2016 IEEE Conference on Visual Analytics Science and Technology (VAST)
23-28 October, Baltimore, Maryland, USA
978-1-5090-5661-3/16/\$31.00 ©2016 IEEE

with a student review manager and three graduate students.

Keywords: Temporal event sequences, recommender systems, prescriptive analytics, visual analytics.

Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Graphical User Interfaces (GUI)

1 INTRODUCTION

The growing interest in event analytics has resulted in a flurry of novel tools and applications using visual analytics techniques to tackle varied problems in healthcare, customer service, education, cybersecurity, etc. The central tasks include describing, summarizing, or comparing collections of event patterns, searching event sequences to find records of interest or build cohorts, predicting outcomes associated with event patterns, studying variants from established workflows, etc. We believe the next breakthroughs for event analytics will come by going beyond the usual descriptive and predictive analytics to develop actionable guidance by way of prescriptive analytics [16, 24].

In layman's terms, the prescriptive analytics for temporal event sequences consists of recommended actions (what and when) that

would lead to the desired outcome based on the history of similar archived records. Imagine the following scenario: I am a student at the end of my second year of graduate school. I wish to become a professor and wonder what jobs other students like me got. Then, I wonder what those who ended up being professors did in their last two years of studies. Did they go on internships? When and how many times? I know that publishing is important, but when did they typically publish papers? Does it seem better to start early or all at the end? Did they get a masters on the way? Did they work as teaching assistants? Early on or later toward the end? So I meet with my department's graduate advisor. He pulls a set of students' records from the campus archives who are similar to me based on their first two years of studies. He explains to me their outcomes in terms of the time it took to graduate and job type. Then, we look at those who became professors, review the recommendations, and discuss together an action plan, combining the wisdom of the advisor and the system's recommendations based on events and timings identified as correlated with becoming a professor.

The research question is *what combination of algorithmic analysis and interactive visual exploration can augment analysts' ability to review recommended actions and improve outcomes?*

Recommender systems are being widely used to assist people in making decisions, for example, recommending films to watch or books to buy. The main novelty of the approach proposed in this paper is that it uses event sequences as features to identify similar records and provide appropriate recommendations. While traditional product recommendations can be described with simple explanations such as "customers with attributes like yours also looked at this product or watched this movie," our approach can be summarized by the following statement: "Based on what happened to customers who started with an event sequence similar to yours, what the sequences of actions and their timings are that might lead to your desired outcome."

Properly presenting and explaining recommendations is critical to the effectiveness of recommender systems and decision support tools in general, as it helps develop users' trust in the system and motivate users' actions [34]. Visualization techniques, such as ranked lists [44] and two-dimensional maps [13], have been used to pursue this goal. EventAction provides a visual analytics approach to (1) find similar archived records, (2) explore potential outcomes, (3) review recommended temporal event sequences that might help achieve the users' goals and identify key steps that are of particular importance, and (4) assist users as they interactively define a personalized action plan associated with a probability of success. The main contributions of this paper are as follows:

- The first attempt—to the best of our knowledge—at a prescriptive analytics system to present and explain recommendations of temporal event sequences.
- A proposed four-step workflow for temporal event sequence recommendation.
- A design study of EventAction, which instantiates the proposed workflow in the context of a student advising application, and reports on an evaluation conducted with a student review manager and three graduate students.

The general EventAction principles instantiated in the student advising application can be applied to many other domains. In the case of doctors formulating medical treatment plans, EventAction can help doctors find archived patients who have medical histories similar to the current patient and identify treatments associated with a good outcome. Another application might be eCommerce companies planning a series of interventions to retain a current customer. They would find archived customers who started with an event sequence similar to the current customer, and then recommend sequences of actions and their timings that increase the likelihood of

retention. A third promising domain is sports coaching. For example, in the middle of a basketball game, a good coach formulates a plan to increase the team's likelihood of winning the game. EventAction can help the coach find archived games that had a similar first half, and suggest actions such as using an agile point guard immediately or attempting more three-pointers in the last five minutes.

2 RELATED WORK

This section discusses related work in event sequence visualization and query, outcome analysis, and recommender systems.

2.1 Temporal Event Sequences Visualizations

Early research on visualizing temporal event sequences focuses on showing individual records. For example, LifeLines [29] and Lifelines2 [41] place events on a horizontal timeline to show when the events occurred. Episogram [10] draws vertical threads on top of a horizontal timeline to represent events that belong to specific conversations or topics. These techniques are capable of showing the detailed events of each record but do not scale well when a large number of records are to be shown in a stacked manner.

Techniques for generating an aggregated overview of multiple records have been designed to tackle this challenge. LifeFlow [43] aggregates multiple event sequences into a tree structure and OutFlow [42] summarizes multiple event sequences as a network. DecisionFlow [18] introduces a set of query and milestone based methods for analyzing event sequences with larger numbers of event categories. Bernard et al. [5] demonstrates how a customized application can provide useful summaries of patient histories and their outcomes, and facilitate the selection of similar patients based on patient attributes.

Our prototype consists of both timeline views for showing detailed events of individual records and activity summary views for revealing event patterns of a group. Our designs were inspired by prior work and adapted to the needs of presenting and explaining temporal event sequence recommendations.

2.2 Temporal Event Sequence Queries

Tools have been developed to help users specify temporal queries, which consist of elements such as the required events, temporal relationships between the events, and attribute ranges of the events or records [22, 27, 35]. The results are event sequences that exactly match the query, which requires users to have specific query rules in mind to obtain useful results. These tools also provide visual feedback to facilitate the iterative refinements of the queries.

Another tool, Similan [44], focuses on searching for similar event sequences. For example, given an event sequence, find other event sequences that are similar to it. It defines similarity metrics to compare two event sequences and takes in consideration of swaps, missing or extra events, and difference in timing between events. The output is a ranked list of the similar records. Users do not need to specify the query rules but the similarity scores are hard to interpret and using the control panels to adjust parameters is complex.

Finding users similar to the active user is a major component of recommendation techniques [32, 37] and has also been applied in other domains such as the similarity-based data-driven forecasting for time series [8]. Our work borrows and extends the existing similarity metrics for comparing temporal event sequences.

2.3 Outcome Analysis

Understanding how different sequences of events lead to different outcomes is an important task in event sequence analysis, leading to hypotheses about causation. OutFlow [42] uses a network structure to aggregate similar event sequences into progression pathways and summarizes the pathways' possible outcomes. Its application for electronic medical records, CareFlow [28], allows doctors to

analyze treatment plans and their outcomes for patients with certain clinical conditions. TreatmentExplorer [17] provides a novel graphic interface for presenting the outcomes, symptoms, and side effects of treatment plans. CareCruiser [19] enables doctors to retrospectively explore the effects of previously applied clinical actions on a patient's condition. CoCo [25] helps analysts compare two groups of records (e.g., with different outcomes) and uses high-volume hypothesis testing to systematically explore differences in the composition of the event sequences found in the two groups. MatrixWave [45] allows the exploration and comparison of two sets of event sequences with different outcomes by displaying the event sequences in a matrix and showing their differences at each step.

These tools visualize the outcomes of a given set of records, enabling users to see the outcomes and progression pathways associated with these records. Our approach is to extend these work by providing recommended sequences of temporal events that might help achieve users' desired outcomes. It also allows users to define personalized action plans and provides feedback on the probability of success. In addition, while most existing tools assume a binary outcome, our approach enables users to explore multiple outcomes.

2.4 Recommender Systems

Recommender systems are software tools and techniques that suggest items for a user [32]. Existing recommendation techniques can be categorized into six classes [9]: *Content-based*, which recommends items similar to what the users liked in the past; *Demographic*, which personalizes suggestions based on the user's demographic attributes such as age or country; *Collaborative Filtering*, which finds other users with similar tastes and recommend items they liked to the active user (e.g., GroupLens [31], Amazon.com [23], item-based algorithms by Sarwar et al. [36], and an empirical study by Herlocker et al. [20]); *Knowledge-based*, which relies on specific domain knowledge to recommend items to meet the user's needs (e.g., case-based recommender systems [6, 33]); *Community-based*, which crowdsources the user's personal social networks for recommendations (e.g., Ben-Shimon et al. [4] and Arazy et al. [3]); *Hybrid Recommender Systems*, which combine the other five recommendation approaches (e.g., Claypool et al. [11] and Mobasher et al. [26]).

An important application domain of recommender systems is education. Educational recommender systems have been developed to provide individual learners with suitable learning resources. Adaptive hypermedia systems [7, 40] suggest learning materials that accommodate each learner's needs to support an active and self-regulated learning. Learning networks [14] connect distributed learners in certain domains and record their learning activities with measures like time and learning outcomes. Learners can use the networks to identify learning paths that are faster to complete or have a better outcome than others. Cognitive tutors [2, 12] recommend appropriate problem-solving activities and provide personalized instructions based on each learner's learning progress to guide the development of problem-solving skills.

We propose a prescriptive analytics approach designed to present and explain recommendations of temporal event sequences. Our prototype extends the Collaborative Filtering technique and recommends actions by referring to archived records that shared similar event sequence patterns with the current record and had the desired outcome. It also augments traditional educational recommender systems by guiding users to define a personalized action plan associated with an increased probability of success.

3 DRIVING APPLICATION AND NEEDS ANALYSIS

The new concept of EventAction had been germinating in our team for several months based on prior event sequence analytics case studies. The design process was accelerated by choosing a specific application (student advising) to drive a multi-phase design study.

Our process was inspired by the nine-stage framework proposed by Sedlmair et al. [38]. Specifically, our work roughly matches the *learn* (visualization literature), *discover* (tasks and needs), *design* (visual, interaction, and algorithm), *implement* (prototypes), *deploy* (to domain expert and gather feedback), *reflect* (on designs and refine guidelines), and *write* (design study paper) stages in that framework. This section focuses on the discover stage, while later sections cover the design, implement, deploy, and reflect stages, which informed revisions to the user and task characterizations, and led to refinements to the prototype.

To learn about student academic planning, we worked closely with the professor who manages the computer science department's review of graduate student progress and has eleven years of experience in student advising. We will call this main category of target user the "review manager." The department conducts annual reviews of students' accomplishments to encourage progress through program milestones. Students report their activities during the past year, including the series of courses they took, papers they published, internships, awards, etc. Based on these temporal event sequence data, the review manager conducts one-on-one reviewing sessions with the students to provide recommendations and help them plan the subsequent years so they may reach their career goal.

Often, the review manager makes recommendations by referring to the department's requirements and by recalling the experience of students he advised in the past. While certain general recommendations such as "*finishing your classes no later than the fifth semester*" or "*starting to work with professors in the second year*" can be made in this manner, the review manager found it difficult to personalize the recommendations to fit each student's progress and career goal, and finding relevant stories from past student histories that may provide inspiration and encouragement.

Facing this challenge, the review manager needs a tool to help him analyze the collected dataset of archived students' academic activities, and augment his ability to make personalized recommendations for each student. We held weekly meetings with the review manager during which we conducted informal interviews to understand the advising workflow and demonstrated the early prototypes of EventAction to collect his feedback and suggestions. Based on the discussions, we gathered and refined a list of design needs that EventAction should support to augment the advising workflow:

- N1. Find Similar Archived Students:** Querying the archived students' data to find those whose activities are similar to the current student in their early years in school.
- N2. Estimate Potential Outcomes:** Summarizing the outcomes of the similar archived students to estimate the outcome of the current student.
- N3. Recommend Actions:** Providing recommendations on what actions to take and when to take the actions to improve the current student's likelihood of achieving the desired outcome.
- N4. Evaluate Action Plans:** Providing immediate feedback on the action plan made by the current student and enabling the current student to review and tune the action plan iteratively based on the feedback.
- N5. Protect Privacy:** Protecting students' privacy by showing only safe aggregations and providing adequate management of access rights to the detailed information.

We identified three variant scenarios of use: (1) the review manager might use the tool independently, for example, before or after an initial meeting with a student, (2) the review manager might explore the data and review suggestions standing side by side with a student, and (3) a student might use EventAction alone or with a peer. We discuss other usage scenarios in the evaluation and discussion sections.

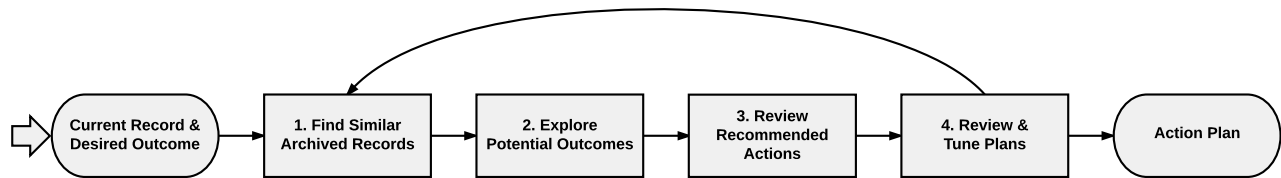


Figure 2: The workflow of EventAction. In the paper, we provide the details of each step using the driving scenario of student advising.

4 DESCRIPTION OF EVENTACTION

EventAction enables a data-driven workflow to help analysts generate a plan of action based on recommendations (Fig. 2). Seeded with a current record for review, EventAction extracts, from the set of all archived records, a cohort of records that are most similar to the current record. Each record is represented as a sequence of events and each event belongs to a particular event category. Outcomes are often defined by the inclusion of certain events in a record, for example, events representing students’ first placements. EventAction estimates the current record’s potential outcomes based on the outcome distribution of the similar archived records, and recommends actions by summarizing the activities of those who achieved the desired outcome. Action plans can be made for the current record and EventAction provides immediate feedback by showing how the plan affects the outcome estimation. In this section, we describe the steps of EventAction’s workflow, using the student advising scenario to illustrate those steps.

4.1 Reviewing Current Record

When using EventAction, a review manager starts by retrieving a current student’s record from the database. The record of a student working alone would be loaded automatically. Users can also select an initial desired outcome. EventAction shows the detail timeline in a table, where each row represents an event category and each column represents a period of time (Fig. 1b). To reduce visual clutter and show periodic patterns, events that occurred during the same time period are aggregated and encoded by the size of the gray square in each table cell. Our initial design was derived from Lifeline2 [41]. It showed the precise timing of all events but caused overlaps when multiple events occur close together. Our revised design applied the bucketing strategy [15] to aggregate the events within time periods, which dramatically simplifies the display.

EventAction allows users to specify the time periods, as they are likely to be highly dependent on specific application domains. For students’ academic records, the review manager segmented each year into three periods according to the school semesters: Spring (January to May), Summer (June to August), and Fall (September to December). The time axis of the current student (Fig. 1b) shows the exact date, while the time axis of the archived students uses relative time (Fig. 1f).

4.2 Finding Similar Archived Records

To find similar archived students, EventAction compares the event sequence patterns of the current student and each archived student. The length of the comparison window is defined by the length of the current student’s timeline. The similarity between two students is measured by the Euclidean distance of the feature vectors extracted from the students’ event sequences within the comparison window. In this paper, we defined the feature vector to be the number of events in each category. We chose a simple similarity algorithm to facilitate our goal of rapidly building a deployable prototype including all the steps of the workflow. The discussion section reviews possible enhancements.

Then, EventAction computes a similarity score between the current student and each archived student and shows the results in the

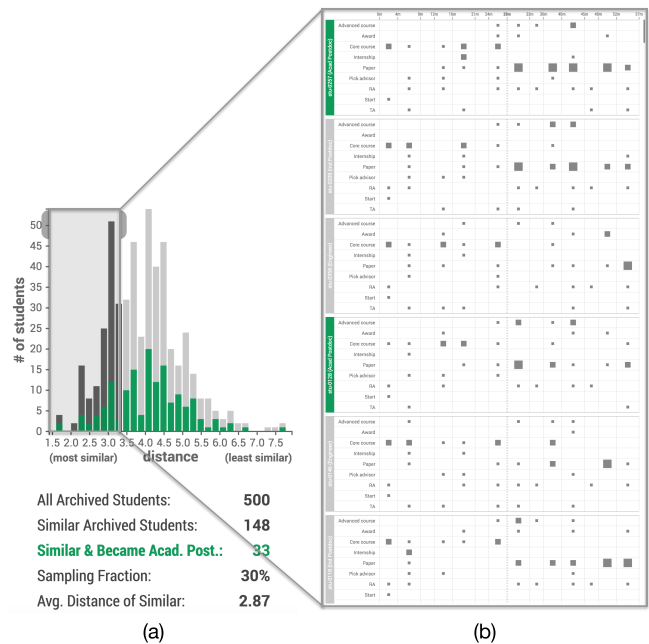


Figure 3: (a) The distribution of the similarity between the current student and each archived student. (b) The timelines of the selected students are displayed for inspection.

similarity distribution view (Fig. 3a). We included a range selection widget to allow users to customize the set of archived students to be considered as the similar cohort. EventAction facilitates the range selection by showing five indicators which were determined through iterative refinement with the review manager: the total number archived students, the number of selected (similar archived) students, the number of selected students with the desired outcome (visible in green), the sampling fraction, and the average similarity score.

After the cohort selection, individual timelines of the similar archived students are displayed for inspection in the lower middle section of the screen, if the user has access rights to those records. (Fig. 3b). The design partner chose to align each record by Fall, which is the typical semester for starting school. Temporal patterns such as the number of courses students take or the most common semester students advance to candidacy become easier to observe.

4.3 Exploring Potential Outcomes

Based on the outcome distribution of similar archived students, EventAction lists the potential outcomes for the current student and estimates likelihoods. The outcome distribution view (Fig. 4) shows two sets of bars: the thicker bars represent the similar archived students, and the thinner bars represent the baseline of all archived students. From this view, users can estimate: (1) the current student’s most likely outcome, (2) the current student’s probability of achieving the desired outcome, and (3) whether the current student

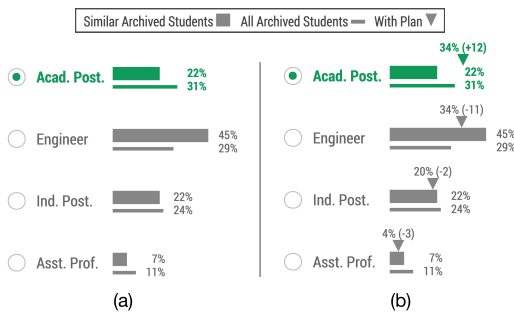


Figure 4: (a) The outcome distributions of similar archived students (thicker bar) and all archived students (thinner bar). (b) EventAction estimates users' action plans and show the updated outcome distribution with triangles. The desired outcome is highlighted in green.

is more or less likely to achieve the desired outcome compared to all archived students. Users can change the desired outcome at any time in the process and all views are updated accordingly.

Using the correlation view (Fig. 5), users can further explore which event categories are most correlated with the probability of having each outcome, so as to identify important event categories that the current student should pay attention to when making the action plan. Each cell or line chart shows the correlation between an outcome and an event category generated based on the similar archived students. The x-axis represents the number of occurrences of that event category in a student's entire timeline. The y-axis represents the probability of having that outcome, which equals to the percentage of students who had that number of occurrences and had that outcome. The size of the dots encodes the number of records. Dots of more than 10 records are connected with lines to show the overall trends. The background color of the charts encodes the Pearson correlation coefficient of the dots, weighted by their sizes. The vertical dashed line shows the number of event occurrences the current student has so far.

Our initial design only used histograms to show the distributions of student populations with different numbers of event occurrences. It was named "feature distribution" but was found not very helpful. Instead of seeing only the distributions, users seemed more interested in learning how the event occurrence is correlated to the probability of achieving an outcome, especially the desired one. Thus, we calculated the percentage values for "probability of success" from the categorical outcome attribute, and added background colors to encode the correlation coefficient. We then changed the histogram to lines and dots to show the detailed relationship between "probability of success" and numbers of event occurrences. To avoid potential misinterpretation, we added text explanations triggered by mouse hovering. We recognize that the correlation information may not be easy for every user to interpret, but its value was immediately recognized by our computer science design partner and students. Simpler designs may be possible.

4.4 Reviewing Recommended Actions

After identifying event categories that are most correlated to the current student's likelihood of achieving the desired outcome, users can explore the activity summary view to investigate the temporal aspect of the recommended actions. Users can choose to show either all or similar archived students (Fig. 6a), and can drill down to see only the activities of those who had the desired outcome of the current record (Fig. 6b), or compare the activities between everyone and those who had the desired outcome (Fig. 6c).

The activity summary view is directly integrated in the timeline of the current record (Fig. 6a) and the activity patterns can be used to guide the specification of the action plan. The background color of each cell in the table represents the percentage of records that

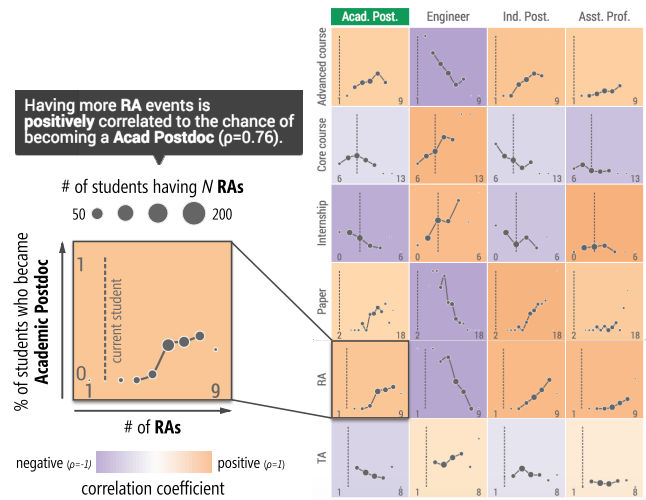


Figure 5: The correlations between outcomes and event categories. The enlarged example chart shows that most of the students had between 4 and 8 RAs, and having more RAs is positively correlated to the current student's likelihood of becoming an Academic Postdoc.

had at least one occurrence of the event category in that time period. The darker the background color, the more prevalent this event category is in this time period. The size of the gray square encodes the most common number of occurrences, which suggests the typical number of this event in this time period. Users can hover on a square to review the detailed distribution of event occurrences.

Our square-based design was inspired by previous work in network comparison [1, 45], which studied different glyph designs for matrix visualizations and found that the square-based method outperformed the rest. Our early prototypes also tried to color the inner square instead of the entire cell. However, this approach makes it difficult to read the color when the square is small. We also considered swapping the mapping, using the background color to represent the number of occurrences and square size to encode the prevalence, but this was inferior to our final design because the visual encoding became inconsistent with the timeline view and our users found the color less precise in representing sparse numbers.

4.5 Reviewing and Tuning Plans

After reviewing the activity summary, users can iteratively specify an action plan with the guidance of the activities of the reference. They can add events of a category and in a time period by clicking on the corresponding cell of the student timeline (Fig. 6d). The planned events are shown as squares side-by-side with the recommended ones and multiple clicks rotate through the range of possible values. The current design was chosen for two main reasons. First, the square-based glyph is simple and consistent with the timeline and activity summary views. Our users were able to understand its meaning immediately. Second, compared to designs that encode only the difference (i.e., where the user plans less or more activities than others), the side-by-side squares give users a more direct overview about the current action plan. It also encourages users to personalize their plan instead of making an "average" plan.

EventAction reruns the workflow to update the outcome estimation periodically (every second by default) as the plan is being updated. Practically, EventAction adds the planned events to the current student's record, extends the comparison window accordingly to the new length of the current student's record, and updates the cohort of similar archived students. Finally, EventAction updates the outcome estimation and shows the changes in the outcome distribution view as triangles (Fig. 4b), giving users immediate feedback on how their action plans affect the estimated likelihood of achiev-



Figure 6: (a) Activities of similar archived records, (b) activities of records that were similar and achieved the desired outcome, (c) activities that distinguished records that achieved the desired outcome (i.e., the difference between (b) and (a)), and (d) users making actions plans with (b) as a reference. The background color of each cell encodes the percentage of records that had at least one event in the period, and the size of the square within the cell shows the typical number of occurrences.

ing the possible outcomes. In this manner, the users can iteratively refine the action until they are satisfied with the results. We chose not to update the views of the similar archived students (lower part of the screen as in Fig. 1e-g) continuously to keep the context stable and focus attention on the outcome estimations.

4.6 Reflections on the Design Evolution

The overall design of EventAction went through a dozen iterations over a three-month period, during which we held weekly meetings with the review manager to deploy and demonstrate the latest version of the prototype, gather his feedback, and discuss an improvement plan. We revised the placement of the seven views of EventAction until the order matched the natural progression of the task. Adding the workflow control panel was very helpful as it suggests the next possible action (e.g., finding similar records or specifying a plan of action) and guides users through the needed steps. Views open as the analysis progresses: only the workflow control panel is open at the start, then the timeline of the selected record can be reviewed, and the similarity distribution view appears, followed by the similar archived record timelines, and the outcome distribution view and correlation view.

Aligning the timelines of the current record and the similar archived records was important, as well as clearly highlighting the time period used for computing the similarity. Again, aggregating

the data by user-specified periods (semesters in this example) both simplified the displays and facilitated the definition of the plan. One important design decision we made was to deliberately avoid suggesting a single recommended series of actions, but instead provide an environment to help users understand the basis for the recommendation and a visual representation of the actions others had taken (like trails in the sand).

Several iterations also led to the consistent use of green color for the desired outcome across different views. Only the correlation view uses a different color palette, mapping a warm orange color hue for positive correlation and cool purple color hue for negative correlation. We made this exception for two reasons. First, if we use green, then only the column that represents the desired outcome should be colored in green while others should not. Thus, we would have to use two color schemes to encode the same information in the same view, which is confusing. Second, the correlation has both negative and positive values. Thus, a bi-color scheme is necessary.

5 EVALUATION

We conducted an exploratory evaluation of EventAction to understand whether and how it was helpful in student advising, and identified its usability issues and limitations. We evaluated EventAction in the three usage scenarios: (1) review manager alone, (2) review manager advising a student, and (3) student making action plans alone. Our evaluation goals were aligned with the workflow of EventAction:

- **Find Similar Archived Students:** Was the meaning of similarity clear? Were there alternative approaches to assessing similarity? What were users' strategies for selecting similar archived students?
- **Explore Potential Outcomes:** Was the outcome estimation based on similar archived students reliable to users? Was the correlation view easy to understand? How would the correlation view assist in making action plans?
- **Review Recommended Actions:** Was the activity summary view easy to understand? Would users be able to identify recommended actions?
- **Review and Tune Plans:** How did users proceed to define their action plans? How often should the outcome estimation be recalculated?

5.1 Review Manager Alone or Advising Students

After several weeks of iterative refinements, the prototype was deployed and made available to our collaborator review manager. He prepared a dataset of 520 archived records of graduated students. Most of the students were enrolled in the PhD program, and their recorded event categories included "start school", "advanced course", "core course", "classes done", "masters degree", "publication", "advanced to candidacy", "TA" (Teaching Assistant), and "RA" (Research Assistant). The review manager categorized the students' first placements into four types, including (1) software engineer, (2) industrial postdoc (e.g., research positions in labs such as Microsoft Research), (3) academic postdoc, and (4) assistant professor. The placement information was used as the students' possible outcomes. The review manager also had access to the records of current students.

The review manager worked on his own computer with a 30-inch display. He was already familiar with the interface so no training was necessary. The entire study consisted of three 2-hour sessions taking place over two weeks. In the following, we describe how EventAction led to a variety of findings and recommended actions.

5.1.1 Exploring All Archived Students

In the first session, the review manager focused on exploring all archived student records to examine the quality of the data and check if the students' performance matched the department's expectation. He chose a random current student and selected all 520 archived students in the similarity distribution view.

At first, he looked at the outcome distribution and correlation views showing the placement information of all archived students, and the activity summary view showing the activity patterns during their studies. He confirmed that the distribution of the students' placements matched his expectation and most of the activities (e.g., courses and assistantships) met the department's requirements.

The hotspots in two event categories attracted his attention: A few students had their "start school" events in the third year instead of at the beginning. The review manager checked the source data and confirmed the pattern, explained by some students being allowed to take classes before being officially admitted. A second finding was that the most common time for advancing to candidacy was the fourth year instead of the fifth (the department's deadline) or the sixth (the effective deadline from the university, after an extension), and he commented that this provided an important insight for improving the department's management, suggesting benefits outside of the one-on-one review scenario.

5.1.2 Becoming an Assistant Professor

In the second session, a third-year Ph.D. student in the department served as the advisee. He described his goal as wanted to become an assistant professor after graduation. The review manager used EventAction to select the top 100 most similar archived students for the analysis.

The outcome distribution showed that the most common outcome of the similar archived students was software engineer and the least common one was assistant professor. Still, the percentage of assistant professors among the similar archived students was higher than that among all archived students. The review manager could easily explain to the advisee the probability of becoming an assistant professor is low but his likelihood was above the average.

Next, the review manager explored the correlation view and looked for event categories that were most positively correlated with the assistant professor outcome, including "publication", "RA", and "advanced course". He noticed that the advisee had already been RA for several semesters but was short of advanced courses and publications. He recommended that the advisee should keep working as an RA, take more advanced courses, and start to accumulate publications.

The review manager then inspected the activity summary view to investigate when might be the best time for these recommended activities. He adjusted the controls to show the aggregated view of the activities of similar archived students who became assistant professors. The results showed a clear pattern of having an RA and publications in each Fall or Spring semester, and that the most common time for taking advanced courses was in the fourth year, before advancing to candidacy. The review manager showed the display to the advisee and they entered a draft action plan together following the pattern. EventAction estimated a 3% increase in the advisee's likelihood of becoming an assistant professor.

The review manager then switched to show activities that distinguished those who became assistant professors from others. Compared to all similar archived students, more of those who became assistant professors had TAs in the final year. The review manager endorsed the benefit of building up teaching experience before going on the job market. They refined the action plan accordingly and the estimated likelihood increased by another 2%.

At the end of this session, the review manager commented: "*Recalling a few memorable prior students and applying [the knowledge] to advise current students is biased. I tend to trust the data*

and statistics." Still, the dialog with the student suggests that the review manager was using his own judgment and experience to evaluate the value of the generated patterns and guide the recommendation process.

5.1.3 Determining an Appropriate Goal

In the third session, the review manager investigated a common situation in which a current student needs help with both determining a goal and making an action plan. He picked a random current student and selected the top 100 most similar archived students. The outcome distribution showed that the current student's likelihood is above the average in becoming a software engineer, but much below the average in becoming an assistant professor. The review manager commented: "*If this student's goal is to become an assistant professor, I would recommend pursuing a postdoc first.*"

The review manager repeated this process and suddenly found an outlier: the student was not similar to most of the archived students as shown in the similarity distribution view. The review manager inspected the student's record in detail and realized that the student made slow progress in both course and research: "*I need to make sure this student knows the department's requirements and deadlines.*" The review manager remarked: "*EventAction could help students get a sense of their situations and help them decide whether to continue their Ph.D. studies or not.*" Future development may also help identify outliers and provide support for reviewing the records before meeting with those students.

5.2 Student Working Alone

In an academic context, to protect the privacy of prior student records and ensure an accurate understanding of the limitations of the data, allowing students to work alone may be infeasible. Nevertheless, we decided to use this scenario as a usability study to guide the design of EventAction. We again use the problem of graduate student advising, but for this test scenario, we constructed a synthetic dataset of 500 archived students, and included features of the real data observed in the study in the previous subsection.

We recruited three current Ph.D. students in our department who had never seen EventAction and elicited their feedback and suggestions. A laptop computer with a 15.4-inch display was used. We asked the participants to imagine that the selected current student was them and to use EventAction to make a plan to increase their likelihoods of achieving their desired outcomes. We provided no training and encouraged the participants to think aloud and report their difficulties and any findings of interest. Each session lasted about 50 minutes. The timeline view showing the records of similar archived students was disabled, just as it would need to be when using real data, in order to protect privacy.

All three participants (referred as *P1-3*) found the workflow control panel easy to use and followed the workflow in their analyses. Below we describe the study results from each step of the workflow.

5.2.1 Find Similar Archived Students

All three participants understood the similarity distribution view and discovered that they could use the selection brush to adjust the cohort of similar archived students. The participants diverged in their strategies for selecting similar archived students. *P1* selected the first half of all archived students as similar and commented: "*The shape looks like a normal distribution so I set the threshold at the average.*" *P2* selected the 100 most similar archived students: "*I only want those who are more similar to me.*" *P3* explored different strategies and decided to set a threshold of a third of the largest similarity score. He explained "*setting a lower bound gives me more confidence.*"

5.2.2 Explore Potential Outcomes

P1 chose academic postdoc as his desired outcome, P2 chose assistant professor, and P3 chose software engineer. All participants verbalized that they could estimate their own likelihoods of having each outcome from the outcome distribution of similar archived students. P1 immediately found that *“my chance seems below the average.”* P2 was concerned about the reliability of the results as he realized that *“the number of assistant professors is small.”* P3 thought the estimation could be more accurate if he could prioritize the event categories and put more weight on core courses. *“These are more relevant to my goal,”* he explained.

All participants had to spend at least five minutes to fully understand the correlation view. One common initial misinterpretation was to see the y-axis of the line chart as the number of students instead of the percentage. P1 and P3 corrected this misinterpretation by themselves as they inspected a few more charts, and the experimenter provided clarification after P2 remained uncertain about the meaning of the correlation chart for five minutes. The participants found many insights after they became familiar with the charts. For example, *“I need to take more advanced courses to increase my chance”* (P1), *“RA and publications are important”* (P2), and *“publications seem not relevant to me”* (P3). P2 and P3 expressed concerns about the large number of charts that need to be inspected. P2 explained *“it is hard to keep track of what I have found, ..., I want a summary statement to remind me of the important things.”* P3 suggested sorting the event categories by their correlations to the desired outcome: *“I want to see the important ones first.”*

5.2.3 Review Recommended Actions

All three participants were able to understand the activity summary view without training. They started by reviewing the activities of both all and similar archived students and found patterns and outliers, such as *“students take more advanced courses than core courses in the later years”* (P1), and *“some students pick advisors as late as in their fourth year”* (P3). They then narrowed down to those who had achieved their desired outcomes. P2 and P3 commented positively on the consistent use of green color for showing data relevant to the desired outcome: *“I know things that are green in the timeline are important and need to pay attention to.”* While P1 and P2 understood the concept of “distinguishable activities,” it took P3 a while to realize it was a simple comparison. *“There are too many levels of subgroups and I was lost,”* P3 explained.

5.2.4 Review and Tune Plans

None of the participants noticed the table cells in the activity summary view became clickable at this step. The experimenter had to provide hints to help them proceed, and P3 suggested providing guidance when users enter this step for the first time. When making action plans, P2 and P3 mainly referred to the activities of those who achieved their desired outcomes, and P2 explained *“I want to at least be similar to these students.”* P1 primarily referred to the activities that distinguish those who became academic postdoc and said: *“These activities can make me stand out from the average.”* All participants used both reference groups and switched between them multiple times. They also referred to the correlation view. *“The correlation view tells me what to do and the activity summary view tells me when to do,”* P3 emphasized.

All three participants explicitly mentioned that EventAction’s immediate feedback made them more motivated to improve the plan: *“I am not satisfied; I probably need to make a better plan,”* P1 said as he found his likelihood of becoming an academic postdoc is still below all archived students. *“The feedback enable me to make and compare alternative plans,”* P3 commented.

In the end, all three participants completed an action plan. P2 was particularly satisfied with the experience and said: *“I appreciated that EventAction is evidence based. It is easier to understand*

than professors’ suggestion. Different professors often gave me different suggestions and confused me a lot.” P1 hoped to make an optimal plan and proposed a feature that *“I only need to set my expectation and EventAction tells me what to do.”* P3 expressed concerns about the reliability of EventAction’s approach that *“the [archived] students might graduate many years ago and things have changed a lot today.”*

In practice, until the accuracy and value of the recommendation and outcome estimation have been validated, it is unlikely that students would interact directly with private data about other students or that students would evaluate the likelihood of outcomes in the absence of guidance and encouragement of an advisor. However, this evaluation step still provided valuable usability information and input from students.

6 DISCUSSION

Our early evaluation suggests that EventAction was helpful for both review managers and students, as they were able to use EventAction to effectively find similar archived students, explore the potential outcomes of the current student, review recommended actions, and prepare and iteratively improve action plans. Overall, the prescriptive analytics workflow of EventAction was easy to learn and the data-driven approach to student advising was appreciated by users.

6.1 Reliability of Recommendations

The holy grail of recommender systems is to convert recommendations into users’ actions. Providing reliable recommendations has the potential to increase users’ trust in the system and thus motivate actions. On the one hand, the reliability depends on the quantity and quality of the data available. To better profile the current advisee and find accurate similar archived records, the data describing each record must be rich, and to find sufficient similar archived records, the data volume must be large and representative. In the early design of EventAction, the data contained only temporal event sequences and outcomes. Additional attributes for records (e.g., demographics information) and events (e.g., the grade of a course) can be included to improve the quality of the retrieved set of similar archived records.

On the other hand, convincing users that the recommendations are actually reliable may prove to be equally difficult, and will require further research on the impact of algorithms and the user interface on users’ perception of the quality of the prediction. Overconfidence can also be an issue. Our users identified several promising elements in the design of EventAction: (1) visually presenting the raw data and the statistics, (2) consistent use of color to mark data and patterns relevant to users’ desired outcomes, (3) providing detailed textual explanations on demand as tooltips, and (4) presenting not only unexpected insights but also expected findings that match the users’ domain knowledge. Users also pointed out several limitations of EventAction. For example, our initial similarity algorithm does not give users the flexibility to tune the similarity measures, and it is difficult to keep track of findings and recall them at the plan-making stage. Besides, although users could open multiple windows to make multiple action plans in parallel, our current prototype does not support saving or visually comparing alternative plans, which would be a useful feature to add.

Compared to the recommendations of products to purchase or films to watch, recommendations using temporal event sequences could yield an exponential number of possible combinations, and differences between two recommended temporal event sequences are likely to be small. The novel approach EventAction uses to solve this problem is that it does not explicitly recommend a particular sequence directly (e.g., Shani et al. [39]), but relies on the user to interpret the probabilities from the correlation analysis and aggregated event sequence on a timeline to construct a reasonably good, even if not optimal, plan.

6.2 Limitations of the User Study

The overall feedback provided by study participants was very positive and generated a lot of discussion and suggestions, nevertheless, we are aware of several limitations. First, the study used a limited number of participants and only computer science students. Working with a wider range of students and majors will inevitably require improvements to the interface. For example, the correlation view may not be usable for some users and alternative designs should be explored—such as textual summaries of the most important results (e.g., a list of the three most important event categories, and general timing insights). Computer science students could discover most features of the interface on their own, but tutorials may be needed for other populations.

Our primary goal was to build a first complete system to demonstrate and start the evaluation of the general approach, but future work will need to focus on improving and validating individual components of the design. Additional steps may also be included, such as checking program requirements to see if they have been met and augmenting recommendations with required steps based on those requirements.

Privacy issues need to be addressed more thoroughly with the implementation of strong safeguards by default but also the exploration of novel strategies specific to this approach. For example, students may grant view access to part of their records to their friends, or the interface may automatically link to publicly available resumes matching the relevant archived students.

Additional improvements might include support for collaboration [21] between advisor and advisee and the use of large displays to instantiate and compare multiple plans of actions [30].

6.3 Scalability and Generality

Scalability remains a challenge for most interactive visualizations. This initial prototype does not tackle scalability issues yet. It runs smoothly with a testing dataset of 10,000 records, each with an average of 42 events. Larger number of archived records would slow down the searching for similar archived students and the automatic re-computation after the action plan is updated. A manual mechanism could be used instead to allow users to decide when to trigger the time-consuming functions. For applications requiring extremely large datasets, such as millions of web customer records, interactive tools using EventAction's current workflow may help researchers understand the role event sequences can play in determining similarity and selecting a plan of action, and ultimately lead to specialized non-interactive algorithms for real-time action selection (i.e., determining a series of interventions).

Finally, the student review application we selected offered useful simplifications allowing the rapid development of a functional prototype that could be deployed for immediate testing. Graduate student records tend to be of similar length, the number of event categories is fairly small, and the semester organization lends itself to a meaningful bucketing strategy to simplify the display of temporal patterns. Easy access to real data and experienced users was also a significant advantage, and our pre-existing familiarity with the general domain and data contributed to our ability to design a useful interface rapidly.

While we believe other application domains will benefit from the general approach of EventAction, further research is needed to tackle the wide variety of event data characteristics and the needs of different users. To start this process we have initiated a collaboration with eCommerce industry partners to investigate the use of EventAction to plan multi-step interventions. Our early discussions suggest that a potential use for EventAction is to help in-house analysts devise and tune the strategies to find similar customers and plan interventions that match the desired outcome (e.g., retain a customer or get him to upgrade) with the goal of later transferring

those strategies to automated algorithms. We have also started investigating applications in the medical domain.

The concept of EventAction was born from our long experience with visual analytics in healthcare, and we believe our approach will provide a fresh way for doctors and researchers to plan long-term medical treatments and follow-up actions associated with a desired outcome. EventAction's approach may facilitate the discussion between patients and medical professionals as they make choices and plan treatment next steps, and—once further refinements are made—may inspire new ways to provide evidence-based medicine and foster patient engagement in the decision process. Our preliminary studies with health data suggest many specific needs. First, interval events (e.g., a week-long hospitalization) need to be treated differently from point events (e.g., a blood test) since the event duration is often critical to making decisions. Passive events (e.g., disease symptoms or diagnoses), which users cannot plan for, should be tagged separately from active interventions (e.g., treatments). Furthermore, users need to be able to prioritize certain events in the records and ignore others—such as those coming from untrusted sources. Finally, records are typically long and complex, so finding a similar case may rely on matching complex patterns but focusing on a small portion of the record.

7 CONCLUSION

The paper described a novel approach for prescriptive analytics that enables analysts to conduct similarity-based data-driven action planning. We designed and implemented a functional prototype called EventAction for a selected application domain (student advising), which was deployed and tested with real student data for a review manager and with synthetic data for three graduate students. Our evaluation demonstrated that the interface could be learned quickly and the proposed workflow was comprehensible. While recommender systems are commonly used, the novelty of our approach is that it uses event sequences as features to identify similar records and appropriate actions. Visual analytics techniques are particularly useful because they provide a rich aggregated presentation of the recommendations, allowing users to explore alternatives and adjust parameters. Analysts can combine prior knowledge and data-driven insights into an actionable plan along with a measure of the likely outcome. To our knowledge, this is the first attempt at a prescriptive analytics interface designed to present and explain recommendations of temporal event sequences. We believe that this approach can be applied to a wide variety of domains such as healthcare or business analytics, and that the paper opens the door to a new direction of promising research.

ACKNOWLEDGEMENTS

We thank all the participants involved in the studies and the reviewers for their valuable feedback. We appreciate the partial support for this research from Adobe.

REFERENCES

- [1] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, and J.-D. Fekete. Weighted graph comparison techniques for brain connectivity analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 483–492, 2013.
- [2] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences*, 4(2):167–207, 1995.
- [3] O. Arazy, N. Kumar, and B. Shapira. Improving social recommender systems. *IT Professional Magazine*, 11(4):38–44, 2009.
- [4] D. Ben-Shimon, A. Tsikinovsky, L. Rokach, A. Meisles, G. Shani, and L. Naamani. Recommender system from personal social networks. In *Advances in Intelligent Web Mastering*, pages 47–55. Springer, 2007.
- [5] J. Bernard, D. Sessler, T. May, T. Schlomm, D. Pehrke, and J. Kohlhammer. A visual-interactive system for prostate cancer cohort

- analysis. *IEEE Computer Graphics and Applications*, 35(3):44–55, 2015.
- [6] D. Bridge, M. H. Göker, L. McGinty, and B. Smyth. Case-based recommender systems. *The Knowledge Engineering Review*, 20(3):315–320, 2005.
- [7] P. Brusilovsky. Developing adaptive educational hypermedia systems: From design models to authoring tools. In *Authoring Tools for Advanced Technology Learning Environments*, pages 377–409. Springer, 2003.
- [8] P. Buono, C. Plaisant, A. Simeone, A. Aris, B. Shneiderman, G. Shmueli, and W. Jank. Similarity-based forecasting with simultaneous previews: A river plot interface for time series forecasting. In *Information Visualization 11th International Conference*, pages 191–196, 2007.
- [9] R. Burke. Hybrid web recommender systems. In *The Adaptive Web*, pages 377–408. Springer, 2007.
- [10] N. Cao, Y. Lin, F. Du, and D. Wang. Episogram: Visual summarization of egocentric social interactions. *IEEE Computer Graphics and Applications*, PP(99):1–9, 2015.
- [11] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin. Combining content-based and collaborative filters in an online newspaper. In *Proceedings of ACM SIGIR workshop on recommender systems*, volume 60, 1999.
- [12] A. T. Corbett, K. R. Koedinger, and W. Hadley. Cognitive Tutors: From the research classroom to all classrooms. *Technology Enhanced Learning: Opportunities for Change*, pages 235–263, 2001.
- [13] J. Donaldson. Music recommendation mapping and interface based on structural network entropy. In *IEEE 23rd International Conference on Data Engineering Workshop*, pages 811–817, 2007.
- [14] H. Drachler, H. G. Hummel, B. Van den Berg, J. Eshuis, W. Waterink, R. Nadolski, A. J. Berlanga, N. Boers, and R. Koper. Effects of the recommender system for navigation support in self-organised learning networks. *Educational Technology & Society*, 12(3):115–126, 2009.
- [15] 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*, PP(99):1–14, 2016.
- [16] J. R. Evans and C. H. Lindner. Business analytics: The next frontier for decision sciences. *Decision Line*, 43(2):4–6, 2012.
- [17] L. Franklin, C. Plaisant, K. Minhazur Rahman, and B. Shneiderman. TreatmentExplorer: An interactive decision aid for medical risk communication and treatment exploration. *Interacting with Computers*, 2014.
- [18] D. Gotz and H. Stavropoulos. Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1783–1792, 2014.
- [19] T. Gschwandtner, W. Aigner, K. Kaiser, S. Miksch, and A. Seyfang. CareCruiser: Exploring and visualizing plans, events, and effects interactively. In *IEEE Pacific Visualization Symposium*, pages 43–50, 2011.
- [20] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53, 2004.
- [21] P. Isenberg, N. Elmqvist, J. Scholtz, D. Cernea, K.-L. Ma, and H. Hagen. Collaborative visualization: Definition, challenges, and research agenda. *Information Visualization*, 10(4):310–326, 2011.
- [22] J. Krause, A. Perer, and H. Stavropoulos. Supporting iterative cohort construction with visual temporal queries. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):91–100, 2016.
- [23] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.
- [24] I. Lustig, B. Dietrich, C. Johnson, and C. Dziekan. The analytics journey [Online]. Available: analytics-magazine.org/the-analytics-journey, 2010.
- [25] 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.
- [26] B. Mobasher, X. Jin, and Y. Zhou. Semantically enhanced collaborative filtering on the web. In *Web Mining: from Web to Semantic Web*, pages 57–76. Springer, 2004.
- [27] M. Monroe, R. Lan, J. del Olmo, B. Shneiderman, C. Plaisant, and J. Millstein. The challenges of specifying intervals and absences in temporal queries: A graphical language approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2349–2358, 2013.
- [28] A. Perer and D. Gotz. Data-driven exploration of care plans for patients. In *CHI Extended Abstracts on Human Factors in Computing Systems*, pages 439–444, 2013.
- [29] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman. LifeLines: Visualizing personal histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 221–227, 1996.
- [30] K. Reda, A. E. Johnson, M. E. Papka, and J. Leigh. Effects of display size and resolution on user behavior and insight acquisition in visual exploration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2759–2768, 2015.
- [31] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 175–186, 1994.
- [32] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [33] F. Ricci, D. Cavada, N. Mirzadeh, and A. Venturini. Case-based travel recommendations. *Destination Recommendation Systems: Behavioural Foundations and Applications*, pages 67–93, 2006.
- [34] F. Ricci, L. Rokach, and B. Shapira. *Recommender systems handbook*, pages 1–35. Springer, 2011.
- [35] A. Rind, T. D. Wang, W. Aigner, S. Miksch, K. Wongsuphasawat, C. Plaisant, and B. Shneiderman. Interactive information visualization to explore and query electronic health records. *Foundations and Trends in Human-Computer Interaction*, 5(3):207–298, 2011.
- [36] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295, 2001.
- [37] J. B. Schafer, J. A. Konstan, and J. Riedl. E-commerce recommendation applications. In *Applications of Data Mining to Electronic Commerce*, pages 115–153. Springer, 2001.
- [38] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440, 2012.
- [39] G. Shani, R. I. Brafman, and D. Heckerman. An MDP-based recommender system. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 453–460, 2002.
- [40] E. Triantafillou, A. Pomportsis, and S. Demetriadis. The design and the formative evaluation of an adaptive educational system based on cognitive styles. *Computers & Education*, 41(1):87–103, 2003.
- [41] T. D. Wang, C. Plaisant, B. Shneiderman, N. Spring, D. Roseman, G. Marchand, V. Mukherjee, and M. Smith. Temporal summaries: Supporting temporal categorical searching, aggregation and comparison. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1049–1056, 2009.
- [42] 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.
- [43] K. Wongsuphasawat, J. A. Guerra Gómez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. LifeFlow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1747–1756, 2011.
- [44] 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.
- [45] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. MatrixWave: Visual comparison of event sequence data. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 259–268, 2015.