

An Evaluation of Visual Analytics Approaches to Comparing Cohorts of Event Sequences

Sana Malik, Fan Du, Megan Monroe, Eberechukwu Onukwugha, Catherine Plaisant, and Ben Shneiderman

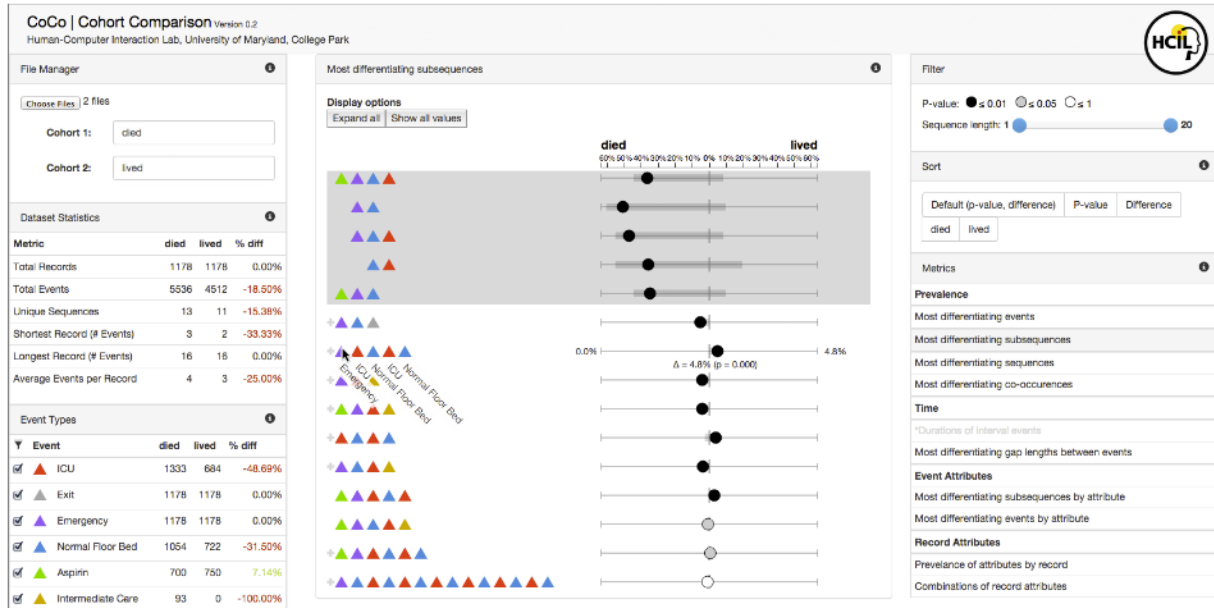


Fig. 1. CoCo combines automated statistical tests with user-guidance to enable insights, hypothesis generation, and much more. In this example, we have two groups of patients as they are transferred throughout a hospital: those who lived and those who died. We can see that being transferred from the emergency room (purple) directly to the normal floor bed (blue) appears statistically significantly more in the group of patients who died ($p \leq 0.01$).

Abstract—A common type of data analysis is finding the differences and similarities between two datasets. With temporal event sequence data, this task is complex because of the variety of ways single events and sequences of events can differ between the two groups (or cohorts) of records: the structure of the event sequences (e.g., event order, co-occurring events, or frequencies of events), the attributes about the events and records (e.g., gender of a patient), or metrics about the timestamps themselves (e.g., duration of an event). Running statistical tests to cover all these cases and determining which results are significant becomes cumbersome. Current visual analytics tools for comparing groups of event sequences emphasize a purely statistical or purely visual approach for comparison. In this paper, we present a novel visual analytics tool, CoCo (for "Cohort Comparison"), which balances automated statistics with user-driven analysis to guide users to significant, distinguishing features between the cohorts. We demonstrate the utility and impact of the visual analytics tool with a user study against a previous visualization-driven approach, EventFlow. For more information about CoCo, visit <http://www.cs.umd.edu/hcil/coco>.

Index Terms—temporal data, cohort comparison, visualization, hypothesis generation

1 INTRODUCTION

Sequences of timestamped events are currently being generated across nearly every domain of data analytics. Consider a typical e-commerce site, tracking each of its users through a series of search results and product pages until, ideally, a purchase is made. Or consider a database of electronic health records, logging the symptoms, medications, and outcomes of each patient who is treated. Every day, this

data type is reviewed by humans who apply statistical tests, hoping to learn everything they can about how these processes work, why they break, and how they can be improved upon.

Human eyes and statistical tests, however, reveal very different things. Statistical tests show metrics, uncertainty, and statistical significance. Human eyes see context, accountability, and most notably, things that they may not have even been looking for.

Visual analytics tools strive to capitalize on these latter, human strengths. For example, the EventFlow visualization tool [19] supports exploratory, visual analyses over large datasets of temporal event sequences. This support for open-ended exploration, however, comes at a cost. The more that a visual analytics tool, such as EventFlow, is designed around open-ended questions and flexible data exploration, the less it is able to effectively integrate automated, statistical analysis. Automated statistics can provide answers, but only when the questions are known.

- Sana Malik, Fan Du, Catherine Plaisant, and Ben Shneiderman are with the Department of Computer Science & the Human-Computer Interaction Lab at the University of Maryland, College Park. Eberechukwu Onukwugha is with the Department of Pharmaceutical Health Services Research at the University of Maryland, Baltimore. Megan Monroe is with IBM Research in Cambridge.

Proc. of IEEE VIS 2014 Workshop on Visualization of Electronic Health Records. Copyright retained by the authors.

Though comparing two groups of data is a common task, with temporal event sequence data in particular, the task of running a multitude of statistical tests becomes increasingly complex because of the variety of ways the cohorts, sequences (entire records), subsequences (a subset of events in a record), and events can differ. In addition to the event sequences' structures (e.g., order, co-occurrences, or frequencies of events), the attributes about the events and records (e.g., patient gender), and the timestamps themselves (e.g., event duration) can be distinguishing features between the cohorts. For this reason, running statistical tests to cover all these cases and determining which results are significant becomes cumbersome. Additionally, the factor on which the cohorts are formed may call for different types of questions to be asked about the data. For example, a set of medical records may be split by date (e.g., last month's trials versus this month's) to see how patient outcomes differ between the cohorts, whereas a dataset split by the patients' outcome (e.g., patients who die versus those who live) would ignore such a metric.

Purely visual tools for temporal event sequences are a good starting point for developing analysis tools for cohort studies, but can be improved by the inclusion of the statistical tests used in automated approaches. This paper introduces CoCo (for "Cohort Comparison"), a visual analytics tool that is designed to capitalize on one such scenario. CoCo is motivated by three years of user studies with EventFlow. While most of these studies began as open-ended explorations, many of them evolved towards one specific objective: cohort comparison. For example, EventFlow assumes that each patient record consists of time-stamped point events (e.g. heart attack, vaccination, first occurrence of symptom), temporal interval events (e.g. medication episode, dietary regime, exercise plan), and patient attributes (e.g. gender, age, weight, ethnic background, etc.). In case studies with EventFlow, we repeatedly observed users visually comparing event patterns in one group of records with those in another group. In simple terms the question was: what are the sequences of events that differentiate one group from the other? A common aspiration is to find clues that lead to new hypotheses about the series of events that lead to particular outcomes, but many other simple questions also involved comparisons. Epidemiologists analyzing the patterns of drug prescriptions [18] tried to compare the patterns of different classes of drugs. Hospital administrators looking at patient journeys through the hospital compared the data of one month with the previous month. Researchers analyzing task performance during trauma resuscitation [4] wanted to compare performance between cases where the response team was alerted of the upcoming arrival of the patient or not alerted. Transportation analysts looking at highway incident responses [11] wanted to compare how an agency handled its incidents differently from another. Our observations suggest that some broad insights can be gained by visually comparing pairs of EventFlow displays (e.g., users could see if the patterns were similar overall between one month and the next) or very different (e.g., a lot more red or the most common patterns were different), but users repeatedly expressed the desire for more systematic ways to compare cohorts of records.

By contrast, CoCo is designed to provide a more balanced integration of both human-driven and automated strategies. Through a preliminary, exploratory user study, we look at how CoCo can support the task of cohort comparison more specifically than previous visualization efforts.

2 RELATED WORKS

2.1 Sequence Comparison

Solutions for comparing sequential data have been explored in many different fields, including comparative genomics, text mining, and tree comparison. They are discussed here in the context of event history data and discrete-time models [1].

We draw first on methods to compare collections of general sequences without the notion of time, most notably the fields of comparative genomics and text mining, where the data is ordered with respect to some index [16].

Genome browsers [6, 9, 14, 22, 24] have been developed to visualize genome sequences. They compare genomes by visualizing the po-

sition of each nucleotide, and consider a genome as a long and linear sequence of nucleotides. Scientists also compare genomes at the gene level. However, most of the existing tools are only able to compare either the similarities or the differences of collections of gene sequences. For example, MizBee [17] measures the similarity between genomes by visualizing the regions of shared sequences. Variant View [8], cBio [5] and MuSiC [7] only support displaying sequence variants.

Texts are often compared by extraction of frequent n-grams [3]. Jankowska et al. [13] proposed to convert documents into vectors of frequent character n-grams and designed a relative n-gram signature to encode the distance between n-gram vectors. Viégas et al. presented history flow [23] to visually compare between versions of a document. Their approach assumes that the later version of a document is developed based on an earlier one, which is not applicable to event histories.

Most of the techniques mentioned above only provide a visual comparison between single long sequences. In our work, the event history data may consist of short transaction sequences, and the comparison supports both visual and statistical approaches.

Temporal event sequences are often represented as trees. While many comparison techniques exist for trees, many do not take into account values or attributes of nodes and none are specifically designed for temporal data. Munzner presented the TreeJuxtaposer [20] system to help biologists explore structural details of phylogenetics and also focuses only on structural differences in the trees. Bremm [2] studied the comparison of phylogenetic trees in a more statistical way. In short, it extended the algorithms of TreeJuxtaposer, and tackled the problem of comparing multiple (much more than two) trees. Holten [12] presented an interactive visualization method to compare different versions of hierarchically organized software systems. TreeVersity2 [10] compares both by tree structure and the node values. Though TreeVersity2 is general to all trees, it leaves out temporal-specific analysis such as duration of interval events. TreeVersity2 compares two datasets from over time, but assumes these time periods are disjoint. CoCo does not assume that the datasets are split by a time attribute and treats time of the nodes as another comparable attribute in the dataset. TreeVersity2 also includes a textual reporting tool that highlights outliers in the data.

2.2 Visualization Tools for Cohort Analysis

EventFlow [19] and OutFlow [25] visualize a simplified view of collections of event and interval sequences. Both tools aggregate a single cohort and the whole sequences of records. EventFlow allows users to explore the underlying dataset through this visualization. However, they only supports visual comparison and does not provide statistics.

Currently tools that combine visualization and statistics for medical cohort analysis focus on single cohorts. CAVA [26] is a visualization tool for interactively refining cohorts and performing statistics on a single group. Recently, Oracle published a visualization tool for cohort study [21]. Based on patients' clinical data, it supports interactive data exploration and provides statistics as well as visualization functionalities. These tools similarly focus on combining visualization with automated statistics and providing an interactive interface for selecting cohorts; however, both tools aim at grouping and identifying patient cohorts for further characterization, while our work focuses on comparing two existing cohorts based on their event histories.

3 COHORT COMPARISON WITH CoCo

With CoCo, users can explore statistics about their datasets using the visualization to organize, sort, and parse the results. After loading files into CoCo, users will see overview statistics about their dataset as a whole, followed by event-level statistics, and lastly a menu of available metrics.

The dataset used as an example for the remainder of the paper consists of records of patients who were admitted to the emergency room and follows their movement through their hospital stay: being administered aspirin, being admitted into the hospital room, transferring between a normal floor bed, intermediate care, and the intensive care unit (ICU), and ultimately being discharged either dead or alive. The patients are split into two cohorts based on whether they lived or died.

3.1 Interface

CoCo consists of a file manager pane, a dataset statistics pane, an event legend, a list of available metrics, the CoCo visualization, and options for filtering and sorting the results (Figure 1).

The summary statistics panel includes high-level statistics about both datasets, including the total number of records and events in each record. Users are then shown the Event Type pane, which serves as the legend (pairing each event type with a marker and color) and filter control. When an event is checked or unchecked, it displays or hides rows containing that event from the analysis. Frequencies of each event type in the two cohorts are also shown.

The right panel consists of filtering and sorting mechanisms and a list of metrics. The metrics are divided by category: prevalence, time, event attributes, and record attributes. Prevalence and attribute significances are calculated by Chi-squared tests and the time significance metrics use a Wilcoxon sum-rank test across the distribution of values. The filtering and sorting panels provide ways for users to parse the results of their selected metric. Users may filter by p-value or sequence length. The default sorting behavior for the results is first by p-value group, then by magnitude of difference. Users may also choose to sort by only p-value, only absolute difference, or by most differentiating in cohort α or by β .

3.2 CoCo Visualization

The CoCo visualization (right in Figure 1) displays the results of significance tests in a unified form. For each event or sequence, CoCo displays the metric value in each cohort (e.g., percent of prevalence, gap duration), the difference between the two values, and the significance value.

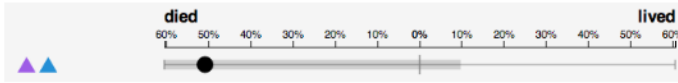


Fig. 2. The sequence “Emergency” followed by a “Normal Floor Bed” event occurs in about 60% of records in “died” and only 10% of records in “lived” for a statistically significant difference of 50% ($p \leq 0.01$)

Each row (Figure 2) consists of a horizontal axis, where the left is cohort α and the right is cohort β . A semi-transparent bar grows from the middle towards each direction in the respective cohort to show the value of the metric for that particular event or sequence. The axis is scaled by the maximum value for all sequences (e.g., if the maximum percentage is 60%, the maximum value on the axis will also be 60%). The circle marker is placed horizontally based on the difference between the values, in the direction of whichever cohort’s value is higher. The circle marker is filled corresponding to the significance of the difference: black is used for p-values with a significance of less than or equal to 0.01, grey for less than or equal to 0.05 but higher than 0.01, and white for values over 0.05 up to 1.

Rows are ranked initially by their p-value (with records with the most significant p-value appearing first), and within each of the three significance groups, rows are ordered by the percent difference between the two groups, from more prevalent in cohort α to more prevalent in cohort β . Users can filter out records by p-values in a certain group with the legend on the top right of the visualization.

4 PRELIMINARY EVALUATION

To inform the refinement and design of CoCo and to observe users’ actual practice of analyzing real-world dataset using a combined CoCo and EventFlow tool, we conducted a preliminary user study with volunteers who expressed interest in learning about data visualization and about a new form of statistical analysis. EventFlow displays an aggregated view of temporal event data, and allows users to easily see the most common event sequences in a group of records.

4.1 Methods

Our evaluation design was based on the VDAR scenario [15]. Our user study goals were (1) to learn about the insights users would find and

(2) to gain insights into the strategies users would follow, both in the context of each tool’s strengths and weaknesses.

Before this study, we tested our materials with four participants, where they used either CoCo or EventFlow and we counted the number of insights. We observed that they tended to report everything from CoCo as insights, without considering their actual meanings and importance. In this study, we asked participants to provide further suggestions based on their insights to guide research at the hospital. As a result, participants were more engaged in the analysis and on average provided 3.5 ($SD = 1.07$) suggestions.

Participants and Settings

We recruited 10 computer science graduate students (7 male, 3 female) through our university’s email list. The participants’ ages ranged from 23 to 29 ($M = 26$, $SD = 2.06$). All participants had normal color vision.

We ran CoCo and EventFlow on the same computer. CoCo was displayed on a 1440×900 screen while two side-by-side EventFlow windows were displayed on a 1920×1200 screen (Figure 3).

For simplicity, we began by implementing only a subset of all possible metrics that users may want:

- Prevalence of events, subsequences, and record sequences.
- Duration between sequentially occurring event pairs.
- Prevalence of events and subsequence by attribute.

Procedure

Each 45-minute session included training, data analysis and post-study interview. Training started with a 2-minute introduction on each interface’s features. For each interface, participants performed 5 simple tasks and were encouraged to ask questions. We used a pair of synthetic datasets for the training. Questions included clarifying the difference between a “sequence” and a “subsequence” (CoCo), the difference between an event type and attribute (CoCo) and the meaning of gaps between bars (EventFlow) were frequently asked. After the training, all participants said they understood everything.

In the 30-minute data analysis session, a different pair of datasets (representative of hospital room transfer data) were used: patients discharged alive or patients who died. Participants were asked to play the role of a data scientist and analyze the datasets using both CoCo and EventFlow. Their job was to provide insights into the similarities and differences between the paths of the two groups in the hospital. We encouraged thinking aloud and an experimenter took notes of their findings. In particular, we asked them to provide a reason when they switched between CoCo and EventFlow.

During the post-study interview, participants provided comments and reflections about their experience.

4.2 Results

During the analysis, no participant asked any interface-related questions and instead concentrated on finding insights. Every participant used both interfaces. In particular, three said they prefer CoCo, while two said they prefer EventFlow. Five expressed no preference. On average, eleven ($SD = 3.67$) insights were reported and four ($SD = 1.12$) interface switches were made per participant. During the interview, all participants stated they wanted to use both interfaces. Below we summarize the results in the context of our user study goals.

Categorization of Insights

All insights can be categorized into four categories: events, whole record sequences, subsequences and time (Figure 4).

Seven out of the ten participants mentioned that it is easier to find subsequence patterns with CoCo while it is easier to find whole record sequence patterns with the side-by-side EventFlow display. They thought EventFlow didn’t actually support detecting subsequence patterns because it only showed records as sequences, and they had to visually scan each record to compare subsequences. On the other hand,

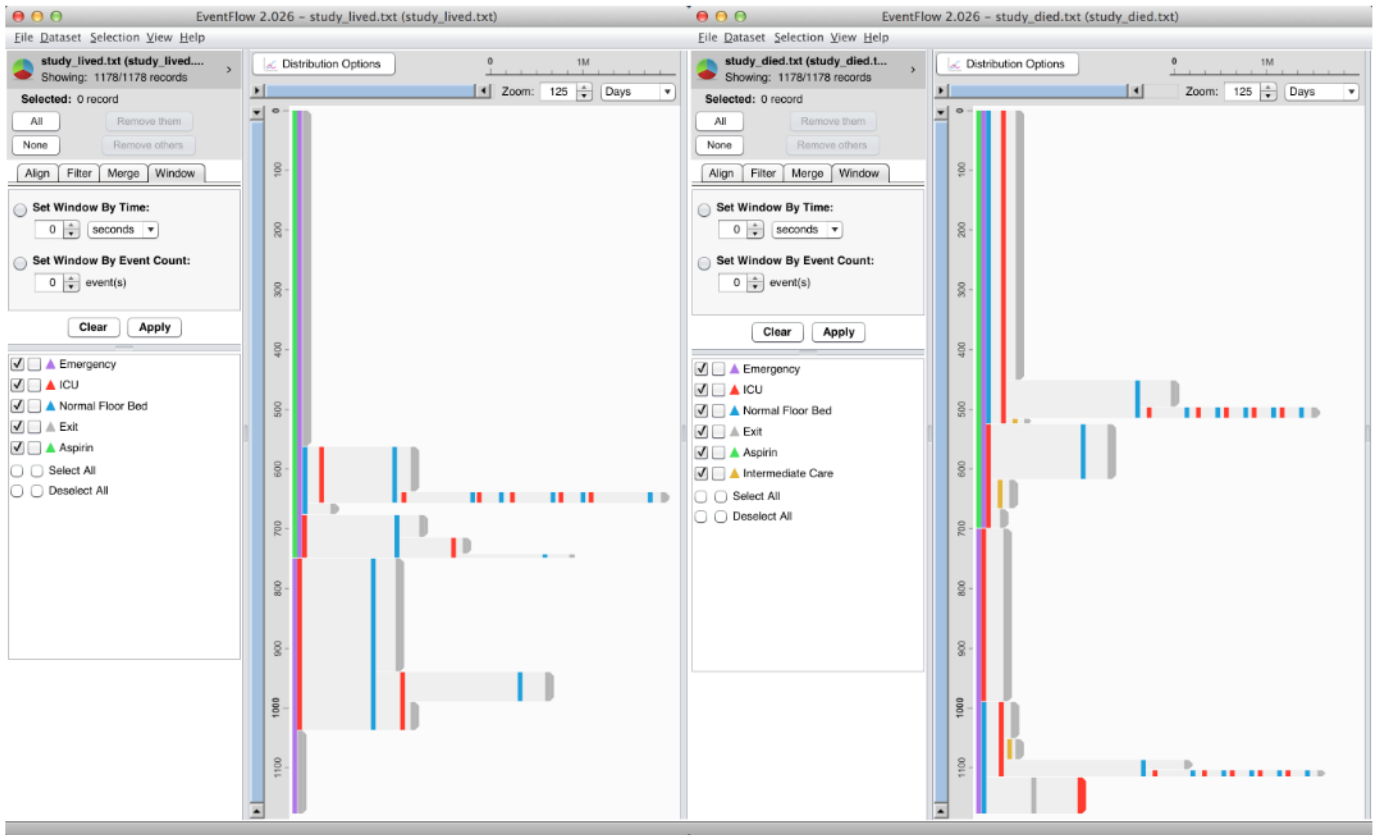


Fig. 3. To compare cohorts in EventFlow, we displayed two windows (one for each cohort) side-by-side on a 1920 × 1200 resolution screen. Real-world users commonly used this technique when comparing two sets of data.

CoCo specifically provided a metric for subsequences. As a result, significantly more subsequence insights were found by using CoCo than EventFlow ($p < 0.05$) (Figure 4).

As for whole record sequences, they preferred EventFlow and stated that it visually encoded the number of records into the heights of color bars, which made it more comprehensive, e.g., “I just have no feeling about the numbers in CoCo. EventFlow is more interesting.” Meanwhile, six out of the ten participants mentioned that it was hard to compare a specific sequence using the side-by-side windows of EventFlow because they had to search for that sequence on each side separately and had to visually compare them by height, which was not very accurate. In contrast, a participant noted that “CoCo does the comparison for me.” There are no significant differences between the two interfaces for the number of insights under the whole record sequence category ($p = 0.39$) (Figure 4).

Participants found most of the insights in the event category by using CoCo. This might be related to the fact that CoCo specifically provides a metric for comparing the distribution of events. “It shows numbers and details.” One participant said when he surprisingly found in CoCo that the *Intermediate Care* event only occurred in the group of patients who died. “I didn’t notice that in EventFlow!” he added. Also, as the event metric was listed at the top of the metric list in CoCo, all participants started using CoCo by looking at that metric.

As for the time metric, six out of the ten participants mentioned that they liked the way EventFlow combined gaps with sequences together. They also mentioned that in CoCo, they had to switch back and forth between the time metric and the sequence metric to look for insights. “It provides a better big picture and the time is more visible,” one participant commented when he was looking at EventFlow to help himself understand the time gaps in the dataset. More insights in the time category were found by using CoCo but it was not significant ($p = 0.25$). One potential reason might be that CoCo was able to average the gap between event pairs, while those pairs were more difficult to find in

EventFlow since they are not aggregated and could appear in multiple places in the visualization.

Description of Strategies

During the study, participants were allowed to switch freely between the two interfaces, but we asked participants to describe why they switched.

Nine out of ten participants chose to start with EventFlow. Of these nine participants, three said EventFlow provided a better overview, four said EventFlow seemed simpler and more comprehensive than CoCo (e.g., “EventFlow is more friendly to my eyes!”), and the other two said they wanted to look at the actual data with EventFlow before doing the analysis. The only participant who chose to start with CoCo said he liked the statistical summary provided by CoCo. “It looks like a dashboard,” he added.

After exploring for five to fifteen minutes, the nine participants who had started with EventFlow switched to CoCo. Three spent about fifteen minutes before making the switch. They explained that they preferred to find everything with one interface first. Four said they got stuck with EventFlow and wanted to get some inspirations from CoCo. Two said EventFlow didn’t support comparing subsequences very well, so they temporarily switched to CoCo to do the comparison. The only participant who started with CoCo also switched to EventFlow. He mentioned that he had found an interesting pattern so he switched to EventFlow to have a look at the actual data, which might help to confirm the finding.

After the first switch, all participants were familiar with both interfaces and on average made three more switches. The reasons they provided mainly fell into three categories: (1) switching from CoCo to EventFlow for the overview of event sequences and gaps, (2) switching from CoCo to EventFlow to “get a sense” of the patterns they found in CoCo, and (3) switching from EventFlow to CoCo to see the statistical information and to confirm their findings. “I like the fact

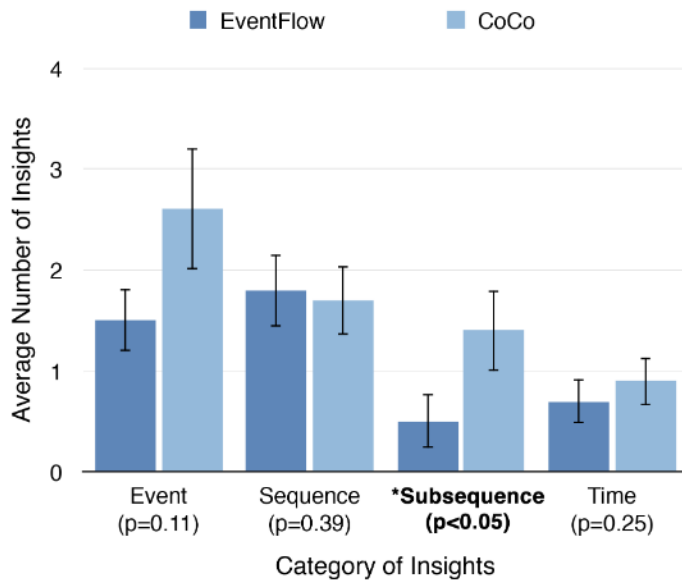


Fig. 4. Average number of insights per participant per category using EventFlow versus CoCo. The only statistically significant difference ($p < 0.05$) is in insights about subsequences, where participants found more insights using CoCo.

that you give me two different tools. I can look at the data in different ways,” one participant commented at the end of his analysis session.

5 CONCLUSIONS AND FUTURE WORK

CoCo is a novel visual analytics tool with a wide array of applications, including facilitating collaboration between colleagues, discussing intermediate results of data, or presenting conclusions in a meaningful way, though it can be extended in many ways. Our current focus has been sequential subsequences in each record, but it will be natural to generalize to any co-occurring events, not just those which occur one after the other, e.g. did patients have more than three aspirins at any time during their treatment. The comments by participants in the preliminary user study and the way they compared the cohorts motivate design changes to the current implementation of CoCo, as well the inclusion of completely new displays. Results of the user study also motivate the need to facilitate more attribute exploration. Additional data mining and statistical techniques could be added to improve insight discovery, such as anomaly detection to find unusual records or clustering find similar records between the datasets.

While we are encouraged by our initial feedback, we see a huge set of possible features to add to aid medical and other researchers as they conduct exploratory data analysis on temporal event sequences.

ACKNOWLEDGMENTS

The authors wish to thank Chris Imbriano and Seth Powsner for their input during the development of CoCo. We also thank our partners at the School of Pharmacy at University of Maryland, Baltimore. This work is supported by Oracle.

REFERENCES

- [1] P. D. Allison. Discrete-time methods for the analysis of event histories. *Sociological Methodology*, 13(1):61–98, 1982.
- [2] S. Bremm, T. von Landesberger, M. Hess, T. Schreck, P. Weil, and K. Hamacher. Interactive visual comparison of multiple trees. In *Proc. 2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 31–40, 2011.
- [3] P. F. Brown, P. V. DeSouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, Dec. 1992.
- [4] E. Carter, R. Burd, M. Monroe, C. Plaisant, and B. Shneiderman. Using eventflow to analyze task performance during trauma resuscitation, 2013.
- [5] E. Cerami, J. Gao, U. Dogrusoz, B. E. Gross, S. O. Sumer, B. A. Aksoy, A. Jacobsen, C. J. Byrne, M. L. Heuer, E. Larsson, Y. Antipin, B. Reva, A. P. Goldberg, C. Sander, and N. Schultz. The cBio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data. *Cancer Discovery*, 2(5):401–4, May 2012.
- [6] Y. Chen, F. Cunningham, D. Rios, W. M. McLaren, J. Smith, B. Pritchard, G. M. Spudich, S. Brent, E. Kulesha, P. Marin-Garcia, D. Smedley, E. Birney, and P. Flicek. Ensembl variation resources. *BMC genomics*, 11(1):293, Jan. 2010.
- [7] N. D. Dees, Q. Zhang, C. Kandoth, M. C. Wendl, W. Schierding, D. C. Koboldt, T. B. Mooney, M. B. Callaway, D. Dooling, E. R. Mardis, R. K. Wilson, and L. Ding. MuSiC: identifying mutational significance in cancer genomes. *Genome Research*, 22(8):1589–98, Aug. 2012.
- [8] J. A. Ferstay, C. B. Nielsen, and T. Munzner. Variant view: visualizing sequence variants in their gene context. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2546–55, Dec. 2013.
- [9] M. Fiume, V. Williams, A. Brook, and M. Brudno. Savant: genome browser for high-throughput sequencing data. *Bioinformatics (Oxford, England)*, 26(16):1938–44, Aug. 2010.
- [10] J. A. Guerra-gomez, M. L. Pack, C. Plaisant, and B. Shneiderman. Visualizing changes over time in datasets using dynamic hierarchies. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2566–2575, 2013.
- [11] J. A. Guerra-gomez, K. Wongsuphasawat, T. D. Wang, M. L. Pack, and C. Plaisant. Analyzing incident management event sequences with interactive visualization, 2011.
- [12] D. Holten and J. J. van Wijk. Visual Comparison of Hierarchically Organized Data. *Computer Graphics Forum*, 27(3):759–766, May 2008.
- [13] M. Jankowska, V. Keselj, and E. Milios. Relative N-gram signatures: Document visualization at the level of character N-grams. In *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 103–112. IEEE, Oct. 2012.
- [14] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and a. D. Haussler. The Human Genome Browser at UCSC. *Genome Research*, 12(6):996–1006, May 2002.
- [15] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536, Nov. 2011.
- [16] S. Laxman and P. S. Sastry. A survey of temporal data mining. *Sadhana*, 31(2):173–198, Apr. 2006.
- [17] M. Meyer, T. Munzner, and H. Pfister. MizBee: a multiscale synteny browser. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):897–904, Jan. 2009.
- [18] M. Monroe, T. E. Meyer, C. Plaisant, R. Lan, K. Wongsuphasawat, T. S. Coster, S. Gold, J. Millstein, and B. Shneiderman. Visualizing patterns of drug prescriptions with eventflow: A pilot study of asthma medications in the military health system, 2013.
- [19] M. Monroe, K. Wongsuphasawat, C. Plaisant, B. Shneiderman, J. Millstein, and S. Gold. Exploring Point and Interval Event Patterns : Display Methods and Interactive Visual Query. Technical Report Technical Report HCIL-2012-06, HCIL, University of Maryland, College Park, Maryland, 2012.
- [20] T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, and Y. Zhou. Tree-Juxtaposer: Scalable Tree Comparison using Focus+Context with Guaranteed Visibility. In *ACM SIGGRAPH 2003*, number 1, page 453, New York, New York, USA, 2003. ACM Press.
- [21] Oracle. Oracle Health Sciences Cohort Explorer User’s Guide. Technical report, Oracle, 2011.
- [22] H. Thorvaldsdóttir, J. T. Robinson, and J. P. Mesirov. Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. *Briefings in Bioinformatics*, 14(2):178–92, Mar. 2013.
- [23] F. B. Viégas, M. Wattenberg, and K. Dave. Studying cooperation and conflict between authors with history flow visualizations. In *Proc. 2004 Conference on Human Factors in Computing Systems - CHI ’04*, pages 575–582, New York, New York, USA, Apr. 2004. ACM Press.
- [24] J. Wang, L. Kong, G. Gao, and J. Luo. A brief introduction to web-based genome browsers. *Briefings in Bioinformatics*, 14(2):131–43, Mar. 2013.
- [25] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2659–2668, 2012.
- [26] Z. Zhang, D. Gotz, and A. Perer. Interactive Cohort Analysis and Exploration. *Journal of Information Visualization (IVS)*, to appear, 2014.