StageMap: Extracting and Summarizing Progression Stages in Temporal Event Sequences

Fig. 1. Teaser

Abstract— Temporal event sequences becomes increasingly important in many application domains such as website click streams, user interaction logs, electronic health records and car service records. A real world dataset with a large number of event sequences and varying sequence length is complex and difficult to analyze. To support visual exploration of the data, it is desirable yet challenging to provide a concise and meaningful overview of sequences. In this paper, we focus on the stage, i.e., frequently occurred subsequence, which is common in event sequence datasets and carry high level semantics in the data. We present a novel visualization technique to summarize event sequence data into a set of stage progression patterns. The resulted overview is more concise compared with event-level summarization and support level-of-detail exploration. We further introduce StageMap, a visual analytics system with three linked view to visualize the stage-level overview, the event-level patterns and the detailed individual sequences. We also present case studies of using the system in two different domains and discuss advantages and limitations of applying StageMap to various application scenarios.

Index Terms—Time Series Data, Data Transformation and Representation, Visual Knowledge Representation, Visual Analytics

1 Introduction

Temporal event sequences, i.e., ordered series of events which occurred over time, appear in a wide range of domains such as website click streams, user interaction logs in software applications, electronic health records and car service records. Analyzing such data can help yield meaningful insights and therefore support decision making. For example, by analyzing user interaction logs, designers can identify those users who confront usability issues and then design interventions to improve user experience. Similarly, by analyzing the online learning click streams, instructors can better understand the learning behavior of students and improve the course design.

Real world event sequences are often complex. A typical dataset can contain thousands or more distinct sequences with hundreds distinct events. The length of each sequences can vary from a few to hundreds of events. Therefore, many existing visualization techniques are inadequate to directly present the data. Instead of visualizing the raw data, recent works try to visualize the summary of the data with the help of pattern mining models [5, 9–11, 16, 18, 21, 22]. These methods have shown promising results since the data summary has a much lower visual complexity. However, extracting and providing a scalable and meaningful visual summary is still challenging due to the following reasons: First, the existing visual summaries are still not concise enough for large scale dataset. It is desirable to have an overview which itself can support level-of-detail analysis and become even more scalable. Second, traditional pattern mining models only preserve or prioritize statistically significant events and therefore have the risk of misleading users on domain specific tasks. So the summary should keep the detailed information and make sure users are aware of the individual variance within the summary.

In this paper, we propose StageMap, an event sequence summarization method which tries to present sequences with a set of stage progression patterns. Stage progression patterns can be found in many event sequence dataset. For example, in online learning click streams analysis, when a student tries to finish an online assignment, he/she may first browse the assignment and then review a course material or ask questions on the course forum. Since many students follow the same steps to finish the assignment, these sequence of learning activities can

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

be defined as a stage while each activity is recorded as an event. The whole learning behavior of a student can be modeled as a progression of various stages. The benefits of presenting sequences with stages are two-fold: First, since the stage itself can be considered as a summary of events, stage-based summary is in general more concise compared with event-based summary. So it can handle more complex dataset, especially when the length of sequences vary a lot. Second, in many applications, stages contain high-level semantics, so users can easily understand the progression pattern without digging into detailed events.

Our method first extracts a set of frequently occurred stages. We support soft pattern match when extracting stages so that later the individual variance is allowed in the stage progression patterns. We then develop an algorithm to transform the original sequences into a set of progression patterns. Each pattern represents a group of similar sequences and how the corresponding stages evolve over time. As in the online learning click streams analysis, a pattern can show a group of students who share similar learning behavior and show how they gradually learn different topics and finish various course activities. In this paper, a pattern is modeled as a tree structure while each node in the tree is a stage. Other structure such as directed graph can also be applied to present the pattern for different application scenarios. The proposed algorithm can also preserve the hierarchical structure of the sequences so that each pattern can split into several detailed patterns. We then design a visual analytics system to support visual analysis of the summary. The system has three linked views: the stage map view to show the summary of identified progression patterns, the tree view to represent the detailed events for a highlighted pattern and the sequence view to visualize each individual sequences. We further test our method on two real world datasets.

To summarize, the main contribution of this work include:

- A summarized representation of event sequences with a set of stage progression trees as well as an algorithm to transform raw sequences into the summary.
- An interactive visual analytics system which allows users to explore the summary of the data.
- Case studies with real world datasets to demonstrate the effectiveness of the approach.

2 Related Work

This section provides an overview of previous work related to event sequence visualization, including the visual forms for presenting event sequences, the techniques for summarize event sequence and the methods for stage analysis.

2.1 Event Sequence Visualization

The most straightforward way to representing event sequences is placing events along a time axis by their order [8, 17, 31]. Lifelines2 [24] further provides different temporal granularities of the axis to highlight important trends. It is now the most common visual form to represent individual sequences in a visualization system for event sequene analysis.

Other visual forms have also been explored. For example, EventFlow [14, 27], FP-Viz [21], TrailExplorer [19, 20] and CoreFlow [10] model event sequences as a tree structure and visualize it with tree visualization techniques such as Sankey diagram or Sunburst visualization. Besides visualizing the tree-like structure, Sankey diagram has also been used by Outflow [26], CareFlow [15] and DecisionFlow [4] to show the directed graph extracted from event sequences. MatrixWave [32] is a recent work to visualize event sequences with matrix. The matrix visualization can avoid edge crossing in Sankey diagrams and show the relation between each timestamp. Besides, various visual forms can be applied to show additional attributes associated with events, such as using stacked area charts to display the sentiment trends of events [12].

Interaction is also a key component for visual exploration of event sequences. Typical interaction techniques include event alignment [23, 24], sequence query [4, 7, 30] and filtering [2, 3, 28], and etc. Lifelines2 [23, 24] is the early work to support comprehensive event alignment of event sequences. (S|qu)eries [30] utilizes regular expression to support flexible sequence query. Du et al. [2] propose an approach to help identify a group of sequences which are similar to a user selected sequence.

However, these visualization techniques are limited when handling large scale dataset. For more complex dataset, level-of-detail exploration is always required. Most of these techniques are suitable for detailed analysis while we still need overview to support high-level analysis and to guide detailed analysis. Our work also adopts some of the existing techniques but focuses on providing concise and meaningful overview which is unique compared with these techniques.

2.2 Event Sequence Summarization

In Section 2.1, we mentioned there are works model event sequences as tree structure or graph structure. These approaches can be considered as one way to summarize event sequences. The complexity of data is reduced by aggregating same events occurred at the same timestamp. However, these methods are sensitive to individual variance among sequences. Similar sequences may not be aggregated due to small variance, which limits the usage of these methods. Recently, CoreFlow [10] proposes an algorithm to extract the tree structure with only high frequency events which can greatly reduce the size of the tree. The low frequency events are discarded in the resulted visual summary which may lead to missing insights of the data.

Sequence clustering can also aggregate similar sequences and provide overview of data. LogView [13] uses treemap to show the hierarchical clustering results of sequences. Wang et al. [22] propose a technique to support unsupervised clustering and visualize the result with packed circles. Wei et al. [25] uses a self-organizing map to cluster and visualize clickstream data. Many sequence summarization methods can be transformed to a clustering problem, but directly displaying the clusters are not suitable for visual exploration since it is difficult to interpret the meaning of each cluster.

There are also works that generate visual summary based on extracted frequent sequential patterns. TimeStitch [18] applies sequential pattern mining models to medical care data analysis and help users to discover, construct and compare cohorts. Both Frequence [16] and Peekquence [9] use frequent pattern mining algorithm and directly visualize mined patterns to help users analyze the data. Furthermore, a three-stage analytic pipeline [11] has been proposed to identify and prune mined sequential patterns. Recently, Chen et al. [1] propose a two part representation to visualize both the sequential patterns and individual variance with the help of Minimum Description Length principle. In this paper, the concept of progression stage is similar to sequential pattern. However, existing works do not consider the sequential relations among stages which limits the flexibility of presenting the inherit data structure.

2.3 Sequence Stage Analysis

A variety of data mining methods have been proposed to identify stage and its progression while few visualization techniques have been studied. Many of the data mining methods focus on the application domains such as medical data analysis. For example, Zhou et al. [33] use fused lasso formulation for stage detection. Jackson et al. [6] and Wang et al. [12] both detect the underlying stages based on Hidden Markov Model. Yang et al. [29] use EM algorithm to associate stages with sequences. Recently, EventThread [5] visualizes the stage progression patterns with storyline visualization. To the best of our knowledge, it is the first work that focuses on stage progression visualization. The main difference between EventThread and StageMap is that EventThread mainly focuses on visual representation and simply extract stages by uniformly segment sequences, while StageMap try to extract the optimized stages and visually represent the data consistently.

References

[1] Y. Chen, P. Xu, and L. Ren. Sequence synopsis: Optimize visual summary of temporal event data. *IEEE transactions on visualization and computer graphics*, 24(1):45–55, 2018.

[2] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Eventaction: Visual analytics for temporal event sequence recommendation. *Proceedings of the IEEE Visual Analytics Science and Technology*, 2016.

[3] 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.

[4] 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.

[5] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, and N. Cao. Eventthread: Visual summarization and stage analysis of event sequence data. *IEEE transactions on visualization and computer graphics*, 24(1):56–65, 2018.

[6] C. H. Jackson, L. D. Sharples, S. G. Thompson, S. W. Duffy, and E. Couto. Multistate markov models for disease progression with classification error. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52(2):193–209, 2003.

[7] 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.

[8] M. Krstajic, E. Bertini, and D. Keim. Cloudlines: Compact display of event episodes in multiple time-series. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2432–2439, Dec 2011.

[9] B. C. Kwon, J. Verma, and A. Perer. Peekquence: Visual analytics for event sequence data. In *ACM SIGKDD 2016 Workshop on Interactive Data Exploration and Analytics*, 2016.

[10] Z. Liu, B. Kerr, M. Dontcheva, J. Grover, M. Hoffman, and A. Wilson. Coreflow: Extracting and visualizing branching patterns from event sequences. 2017.

[11] Z. Liu, Y. Wang, M. Dontcheva, M. Hoffman, S. Walker, and A. Wilson. Patterns and sequences: Interactive exploration of clickstreams to understand common visitor paths. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):321–330, 2017.

[12] Y. Lu, M. Steptoe, S. Burke, H. Wang, J.-Y. Tsai, H. Davulcu, D. Montgomery, S. R. Corman, and R. Maciejewski. Exploring evolving media discourse through event cueing. *IEEE transactions on visualization and computer graphics*, 22(1):220–229, 2016.

[13] A. Makanju, S. Brooks, A. N. Zincir-Heywood, and E. E. Milios. Logview: Visualizing event log clusters. In *Privacy, Security and Trust, 2008. PST'08. Sixth Annual Conference on*, pp. 99–108. IEEE, 2008.

[14] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal event sequence simplification. *IEEE transactions on visualization and computer graphics*, 19(12):2227–2236, 2013.

[15] A. Perer and D. Gotz. Data-driven exploration of care plans for patients. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pp. 439–444. ACM, 2013.

[16] A. Perer and F. Wang. Frequence: interactive mining and visualization of temporal frequent event sequences. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pp. 153–162. ACM, 2014.

[17] 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*, pp. 221–227. ACM, 1996.

[18] P. J. Polack, S.-T. Chen, M. Kahng, M. Sharmin, and D. H. Chau. Timestitch: Interactive multi-focus cohort discovery and comparison. In *Visual Analytics Science and Technology (VAST), 2015 IEEE Conference on*, pp. 209–210. IEEE, 2015.

[19] Z. Shen and N. Sundaresan. Trail explorer: Understanding user experience in webpage flows. *IEEE VisWeek Discovery Exhibition*, pp. 7–8, 2010.

[20] Z. Shen, J. Wei, N. Sundaresan, and K.-L. Ma. Visual analysis of massive web session data. In *Large Data Analysis and Visualization (LDAV), 2012 IEEE Symposium on*, pp. 65–72. IEEE, 2012.

[21] J. Stasko and E. Zhang. Focus+ context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Information Visualization, 2000. InfoVis 2000. IEEE Symposium on*, pp. 57–65. IEEE, 2000.

[22] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao. Unsupervised clickstream clustering for user behavior analysis. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 225–236. ACM, 2016.

[23] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman. Aligning temporal data by sentinel events: discovering patterns in electronic health records. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 457–466. ACM, 2008.

[24] 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), 2009.

[25] J. Wei, Z. Shen, N. Sundaresan, and K.-L. Ma. Visual cluster exploration of web clickstream data. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pp. 3–12. IEEE, 2012.

[26] 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, Dec 2012.

[27] 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*, pp. 1747–1756. ACM, 2011.

[28] K. Wongsuphasawat and B. Shneiderman. Finding comparable temporal categorical records: A similarity measure with an interactive visualization. In *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pp. 27–34. IEEE, 2009.

[29] J. Yang, J. McAuley, J. Leskovec, P. LePendu, and N. Shah. Finding progression stages in time-evolving event sequences. In *Proceedings of the 23rd international conference on World wide web*, pp. 783–794. ACM, 2014.

[30] E. Zgraggen, S. M. Drucker, D. Fisher, and R. Deline. (s|qu)eries: Visual regular expressions for querying and exploring event sequences. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015*, pp. 2683–2692, 2015.

[31] J. Zhao, C. Collins, F. Chevalier, and R. Balakrishnan. Interactive exploration of implicit and explicit relations in faceted datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2080–2089, 2013.

[32] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. Matrixwave: Visual comparison of event sequence data. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 259–268. ACM, 2015.

[33] J. Zhou, J. Liu, V. A. Narayan, J. Ye, A. D. N. Initiative, et al. Modeling disease progression via multi-task learning. *NeuroImage*, 78:233–248, 2013.