A Survey on Visual Analysis of Event Sequence Data

Yi Guo, Shunan Guo, Zhuochen Jin, Smiti Kaul, David Gotz, and Nan Cao

Abstract—Event sequence data record series of discrete events in the time order of occurrence. They are commonly observed in a variety of applications ranging from electronic health records to network logs, with the characteristics of large-scale, high-dimensional and heterogeneous. This high complexity of event sequence data makes it difficult for analysts to manually explore and find patterns, resulting in ever-increasing needs for computational and perceptual aids from visual analytics techniques to extract and communicate insights from event sequence datasets. In this paper, we review the state-of-the-art visual analytics approaches, characterize them with our proposed design space, and categorize them based on analytical tasks and applications. From our review of relevant literature, we have also identified several remaining research challenges and future research opportunities.

Index Terms—Visual Analysis, Event Sequences, Visualization

1 Introduction

Event sequence data are found across a vast array of applications and domains. In fields as diverse as computer security, advertising, and healthcare, discrete observations of different types are collected over time and arranged in sequences based on the specific entity for which the event is germane. For example, Electronic health records, meanwhile, capture events (e.g., diagnoses, procedures) over time for individual patients. The ubiquity of event sequence data reflects both (1) the relative ease with which it can be captured, and (2) the desire to leverage this form of data to gain new insights about real-world systems.

Formally, a temporal event sequence can be defined as an ordered lists of events: $\vec{s} = [e_0, e_1, ..., e_n]$, where each element $e = (\tau, t)$ represents a distinct event with τ as the event's type and t is the time at which the event occurred. In contrast to time-series data where data are captured in the continuous-time domain with fixed time lags, events in $\vec{s}$ are discrete and can occur at any point in time. An electronic medical record for a patient, for example, can be recognized as a temporal event sequence containing a unique time-sorted list of events from the patient's medical history (e.g., diagnoses, lab tests, medications, and treatments). Even though time-series data can be converted to event sequence by dividing the time-series into time segments and characterizing the data within the segment, in our work, we only focus on works originally proposed for event sequences.

These common goals, however, are challenged by the great heterogeneity that exists within different properties of event sequence data and the types of insights that are sought. For example, event sequences can be high-dimensional (with many event types) or low-dimensional (very few types of events). They can be sparse and irregular over time, or dense and evenly spaced. Events can have zero attributes or many, can be point events or intervals, and can be strictly sequential or occur in parallel. Similarly, the types of analysis tasks can vary widely based on the types of insights one seeks. Are analysts interested in common patterns or rare outliers? Are analysts focused on prediction, or identification of predictive factors for intervention? Are analysts examining a single sequence or comparing across multiple sets of sequences in aggregate? These are just a few examples of the wide variety of data and task challenges which present themselves in event sequence analysis.

These difficult and diverse methodological challenges have motivated a broad range of recent research activities which aim to solve one or more aspects of the event sequence analysis problem. This has led, in turn, to a proliferation of different visual analysis methods and prototypes, each of which has distinct capabilities and advantages in certain contexts. This has resulted in a situation where the state-of-the-art for event sequence data is often difficult to discern. The latest research often offers multiple visual analytics approaches for specific types of challenges. Moreover, the same solution may be effective at addressing difficulties that stem from two or more different challenges. Yet in other cases, open problems remain unaddressed.

The aim of this survey is to provide a comprehensive review and characterization of the state-of-the-art in visual analytics research for event sequence data. Through the collection and analysis of the literature on this topic, we identify key dimensions of the event sequence visual analytics design space. We then use those dimensions, as well as a characterization of different types of event sequence analysis tasks, to organize existing methods and identify common approaches to specific targeted problems. Moreover, we identify areas with little prior work which remain a challenge for future research.

This literature review represents the first (to our knowledge) comprehensive attempt to survey and characterize event sequence data visual analytics methods. In this way, this review promises to help researchers understand key dimensions that unify prior work, how prior research fits together within this complex design space, and which event sequence data analysis challenges remain insufficiently addressed. Moreover, the results can provide value to practitioners as an organized catalog of alternative approaches that are most appropriate for specific types of event se-

- *Yi Guo, Shunan Guo, Zhuochen Jin, and Nan Cao are with the Intelligent Big Data Visualization Lab, Tongji University, Shanghai, China. Nan Cao is the corresponding author. E-mail: nan.cao@gmail.com*
- *Smiti Kaul and David Gotz are with the Visual Analysis and Communication Lab at the University of North Carolina at Chapel Hill, NC, USA. E-mail: smiti, gotz@unc.edu*

quence data problems. We developed a web-based survey browser [1] to facilitate the exploration of our created taxonomy and reviewed techniques.

2 Related Surveys and Methodology

In this section, we first discuss survey papers that are relevant to this work, and then introduce our methodology of selecting papers and creating our taxonomy.

2.1 Related Surveys

This section provides an overview of the surveys that are relevant to the visual analysis of event sequence data. Plaisant et al. [84] proposed a characterization of event sequence data and summarized eight high-level user tasks. Keim et al. [53] proposed a definition and an analytical pipeline for visual analytics, which inspires our formalization of the design space that we discuss later in Section 3. A prior survey by Sun et al. [107] generalized visual analytics techniques according to different data types, among which the review of visual analytics approaches for temporal data is most relevant to our work. Our review, by contrast, focuses on a more specific type of temporal data – event sequence data. In addition, some scholars attempted to dive into particular visual forms or visual analytics approaches for a single analytical task that are partially related to our survey. For example, Brehmer et al. [2] formalized the design space for a representative form for visualization of event sequence data – timeline-based visualizations. Jentner and Keim [47] reviewed visualization and visual analytics methods for exploring frequent patterns. Given the broad application of event sequence data, we also noticed a larger group of surveys linked to applications where event sequences are commonly collected, such as social media data [124], smart manufacturing [140], anomalous user behaviors [99] and health informatics [89]. Different from existing work that summarizes techniques for a particular visualization, visual analytical task, or application related to temporal event sequence, our work aims to provide a more holistic overview of the visual analytics approaches for all types of event sequence data so as to benefit practitioners from a wider range of applications.

2.2 Survey Methodology and Taxonomy

This survey aims to obtain an overview of existing visual analytics techniques that are developed for event sequence data. To make a comprehensive review of existing studies, we collected relevant papers from visualization journals and conferences following two main approaches: reference-driven and search-driven selections. For the reference-driven selection, we utilized a core set of state-of-the-art techniques in this topic known to us in advance as a starting point, and extended the range of work by going through cited and citing publications. For the search-driven selection, we went through two rounds of paper collection. The first round involves a coarse search of event sequence analysis and visualization techniques from high-impact conferences and journals in the field of information visualization and data mining. In particular, we select six visualization conferences (IEEE VAST, IEEE InfoVis, ACM CHI, ACM IUI, EG/IEEE EuroVis, IEEE PacificVis), three visualization journals (IEEE TVCG, IEEE CG&A, Computer Graphics Forum), four data mining conferences (NeruIPS, WWW, ACM SIGKDD, ICML), and two journals (IEEE TKDD, ACM TIST). We used two search queries ("event sequence" `AND` "analysis"; "event sequence" `AND` "visualization") to collect papers broadly, then reviewed the abstracts and full texts to finalize our selection.

To construct a structured and comprehensive taxonomy, we formalized a design space for characterizing each visual analytics approach (discussed in Section 3). In particular, we leverage the conventional visual analytics pipeline [53] that revolves around four key components: data, model, visualization, and knowledge. These four dimensions are interdependent and form a closed loop to structure the whole knowledge discovery process supported through visual analytics. Since deriving knowledge from models and visualizations can be subjective and difficult to standardize, we exclude knowledge inference from the scope of our design space. In addition, user interaction that links the components throughout the pipeline is also indispensable in the visual analytics process. These considerations lead to our final proposed design space with the following four dimensions: *data scales* , *automated sequence analysis*, *visual representations*, and *interactions*. We labeled each work with its corresponding dimension. Note that event sequence analysis techniques are only labeled with one of the first two dimensions. In addition, according to Keim et al. [53], the choices of analysis methods, visual representations, and interactions depend on the analytical tasks and application scenarios. Therefore, we also extracted the motivating analytical tasks and application domains from each technique. This gives us a full list of nine analytical tasks, which we further organized into five categories as outlined in Section 4, and five applications under three major categories as outlined in Section 5. The design space serves as a fine categorization where each technique reflects one or multiple design choices in each dimension. In contrast, the analytical task or application is a coarse categorization that each technique often targets one only.

For each analytical task and application, we went through another round of compliementary paper collection for visualization and visual analytics techniques with search queries that combines specific tasks or applications, such as "event sequence summarization" `AND` "visualization", "medical data" `AND` "visualization", etc. The entire selection process gave us 153 most relevant publications of event sequence analysis and 148 publications of event sequence visualization and visual event sequence analysis. We further refined our selection to 104 most representative and up-to-date event sequence visualization and visual analytics studies to discuss in this paper. Additionally, this survey includes a review of 9 related surveys, 5 event sequence analysis techniques, and 9 visualization techniques in the field of causality analysis yet unrelated to event sequence data, and it refers to 1 book about statistical knowledge, 13 papers regarding the theory of visual analytics, research challenges, and opportunities for visual event sequence analysis. A total of 141 papers are covered in this survey.

The remaining survey is organized as follows. We first propose the design space for characterizing visual analytics techniques in Section 3. Section 4 elaborates on state-of-the-art solutions for each analytical task through an analysis of their corresponding design components within the design space. Then, Section 5 provides an overview of applications where event sequence data are commonly observed, serving as a more direct guide to practitioners of visual analytics. Finally, we discuss research challenges and opportunities in

1. http://eventvis.idvxlab.com/

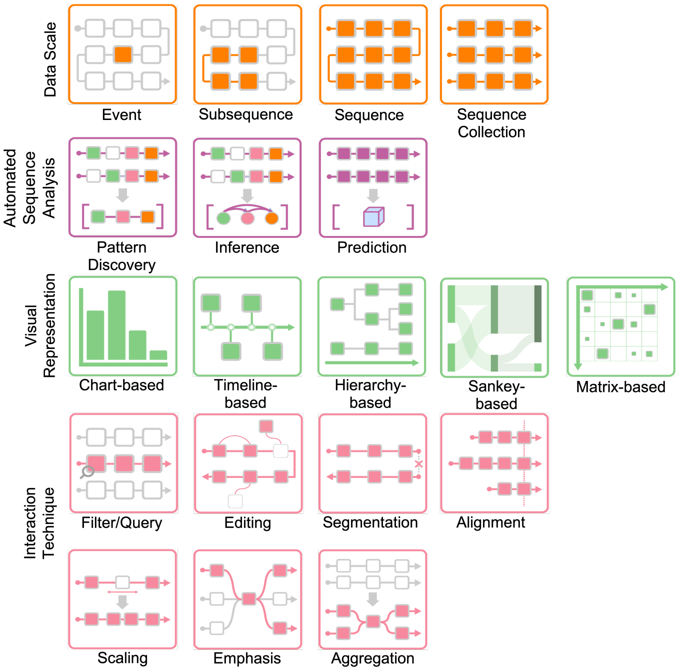


Fig. 1. The design spaces of visual analytics techniques for event sequence data include four dimensions: data scale, automated sequence analysis, visual representation, interaction technique.

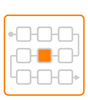
Section 6 and conclude in Section 7.

3 Design Space

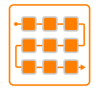
In this section, we propose a design space for characterizing the visual analytics techniques we reviewed.As described in Section 2.2, the four dimensions of design spaces are motivated by the visual analytics pipeline [53], reflecting "data", "model", "visualization" and "user interactions" components in the pipeline, respectively. By classifying existing literature on each dimension, we highlight main categories in each dimension that are frequently utilized for designing and building visual analytics techniques for event sequence data.

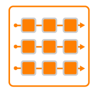
3.1 Dimension 1: Data Scales

Our proposed design space starts by identifying what granularity of data the visual analysis aims to cover. We summarize the following levels of data granularity for any given event sequence dataset.


Event: Individual events represent the finest granularity of event sequences. Each event can be characterized by attributes such as event type, time of occurrence, and duration. Visual analytics techniques often attempt to drill down to individual events to provide users with low-level details of the analysis result. For example, Vistracker [30] identifies anomalous events in trace routes based on event attributes. Carepre [48] predicts upcoming diseases based on the historical sequence of medical events.

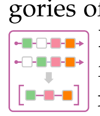

Subsequence: Subsequences are segments of event sequences that preserve the temporal order of events. EventAction [22] uses the number of common subsequences between individuals to measure sequence similarity. MOOCad [73] leverages anomalous frequent subsequences to explore sequence anomalies.

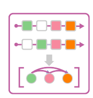

Sequence: An event sequence is the complete record of events that are performed or experienced by a sequence entity (e.g., a patient or a customer). The entire sequence is often analyzed when attempting to get a complete view of the entity's experience. In [38], [77], [138], anomalous entities are detected by analyzing their corresponding progressions of events. Similarly, Guo et al. [36] utilize the embedding of each sequence to estimate the similarity between entities.


Sequence Collection: A collection of sequences are analyzed when summarizing common patterns in the dataset or comparing different groups of sequences. For example, visual summarization techniques [39], [81], [83] aim to provide a summary of patterns and identify entities with common progressions in a collection of sequences. MatrixWave [139] is designed to compare two collections of event sequences and analyze their differences.

3.2 Dimension 2: Automated Sequence Analysis

Visual analytics techniques for event sequence data are incorporated with back-end data mining algorithms to support complex analytical tasks. Based on a review of event sequence analysis methods, we identify the following categories of mining and modeling techniques.


Pattern discovery: Pattern discovery aims to find frequently occurring patterns and statistically significant associations of data samples. In the analysis of event sequence data, pattern discovery techniques can be further categorized into frequent pattern mining techniques and similarity analysis techniques based on different analytical goals. Frequent pattern mining techniques are used to uncover common subsequences in the event sequence dataset. For instance, Perer et al. [81] proposed a visual analytics system that employs a SPAM-based algorithm to extract frequent patterns in a collection of event sequences. Similarity analysis techniques utilize event patterns of each sequence to quantify the similarity between sequences. For example, in [22], [122], two different similarity measurements were proposed based on commonness and differences between events across different event sequences.


Sequence inference: Inference focuses on understanding the relationships between the model's inputs and outcomes according to evidence observed in data [46]. The inference models ensure high interpretability and could explore the effect of a change in input on outcomes. Conclusions derived from inference techniques are tenable under certain conditions but can be incorrect when applied to unobserved data. Existing inference techniques for event sequence analysis mainly include the self-exciting point process and graphical models. The Self-exciting point process is a probabilistic model that describes event occurrence probabilities over time as influenced by historical events. For example, Hawkes Process is widely employed to model sequential data under the assumption that the impact of the previous event can be approximated by a numerical integration over time [69], [128]. Based on the numerical integration, Hawkes Process can infer the relationships between previous and upcoming events. Some graphical models, on the other hand, present the conditional dependence between events with an event correlation graph, such as Bayesian Networks [1] and Markov Chain [103]. According to the correlation graph, models can infer how outcomes will change in response to adjustments in previous events.

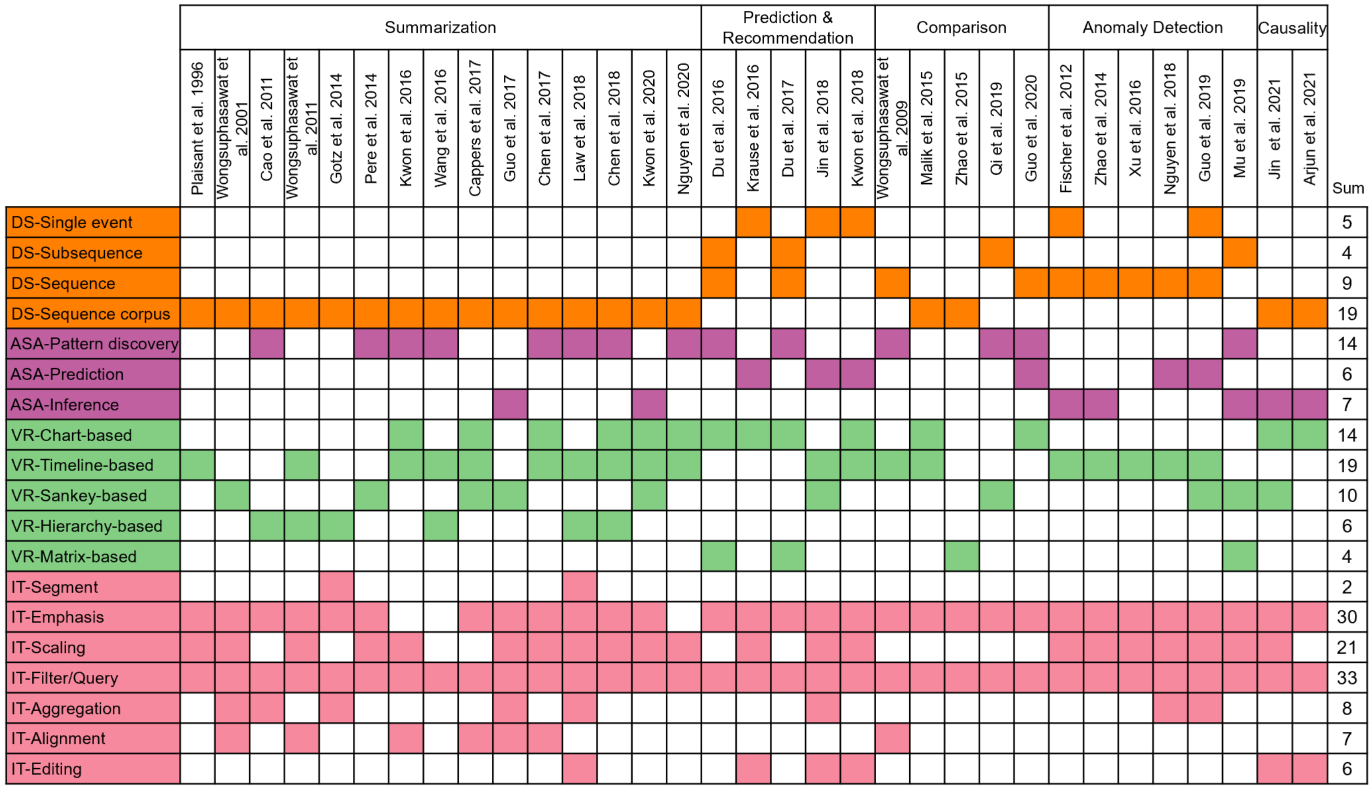
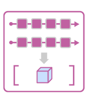


Fig. 2. The most cited papers regarding event sequence visualization and visual analytics techniques grouped by different tasks. Each paper is labeled by the relevant design elements in the design space. The rows are grouped and colored by dimensions of our proposed design space: DSs - Data Scales; ASAs - Automated Sequence Analysis Techniques; VRs - Visualization Representations; ITs - Interaction Techniques.

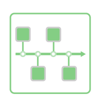


Sequence prediction: While sequence inference techniques are not capable of drawing correct conclusions on unobserved data, sequence prediction methods are developed to build a reliable model which can characterize observed data and ensure the model's generalization abilities to forecast unobserved outcomes simultaneously [46]. To achieve higher accuracy, prediction techniques are often more complex and lack interpretability. Event sequence models are usually designed for specific analytical tasks, such as classification (e.g., kernel support vector machines, decision trees) and clustering (e.g., k-means). Neural network models, especially recurrent neural networks (RNNs), are also commonly used to model event sequences due to their inherent sequential structure and superior performance compared to traditional machine learning models. For instance, CarePre [48] employed attention-based RNNs to predict upcoming events based on historical events in sequences, and Guo et al. [38] embedded RNNs into a Variational Auto-Encoder to detect anomalous sequences in the dataset. However, users are unable to determine the impact of inputs on outcomes when using RNNs that contain numerous non-linear combinations of neurons. As a result, the above methods incorporate visualization techniques to enhance the interpretability of models.

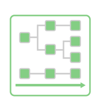
3.3 Dimension 3: Visual Representations

Existing visual analytics techniques leverage a variety of visual representations to display event sequence data and communicate insightful patterns. The visual representations also determine how events and sequences are organized and aggregated. We identify the following five categories of visual representation for displaying event sequence data.

Chart-based visualizations: Visualization charts, such as bar charts and scatter plots, are commonly used to display event features and event distributions in event sequences. For instance, Coco [68] uses a table to compare event distributions of two different groups of sequences and a scatter plot to show the number of records containing particular events or subsequences.

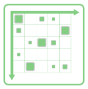


Timeline-based visualizations: Timelines are the most intuitive visualizations that organize events of individual sequences successively in temporal order. Events are generally represented with icons encoded by color, size, or shape to distinguish events with different attributes. For example, VASABI [76] visualizes a sequence as a row of squares colored by event categories.



Hierarchy-based visualizations: In hierarchy-based visualizations, sequences are aggregated into a tree of sequences [121], where each node represents an event placed according to its prefix in the sequence. A variety of visualization designs can be used to display this hierarchical structure of sequences, such as treemaps [108], node-link tree [111], and icicle plots [62], [65].

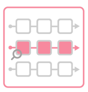
Sankey-based visualizations: Instead of aggregating sequences into a tree structure as the hierarchy-based visualizations, Sankey-based methods organize sequences into the structure of a Sankey diagram [88] and aggregate then into a graph. This technique reveals an overview of transitions that occur between different types of events. Sankey-based visualizations can be further categorized into two different types of design. The first type is the directed node-link graph in which events are represented by nodes and transitions between events are represented by links [39], [48]. The second type is the traditional Sankey diagram, in which links are further encoded by width, representing the proportions of flow that split and merge among events [40], [119].



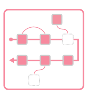
Matrix-based visualizations: Matrix-based visualizations are typically used to demonstrate a summary of event frequency or frequent patterns. For example, EventAction [22] incorporates an event matrix to summarize frequencies of events across different time intervals. Mu et al. [73] applies a matrix-based design to present lists of frequent activity patterns in each stage of sequence progression. In addition, a matrix-based design is also utilized to display frequencies of event transitions. For example, Zhao et al. [139] transforms the traditional Sankey diagram into a sequence of matrices to display step-to-step transitions of web clickstream data.

3.4 Dimension 4: Interactions

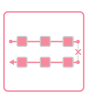
Visual analytics systems usually incorporate rich interactions to empower end users with sufficient flexibility and depth in data analysis. Next, we summarize seven interaction techniques that are commonly applied in visual analytics systems for event sequence data.



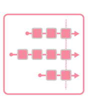
Filter/query allows users to make domain-specific data adjustments or selections based on certain conditions, so as to eliminate noisy and irrelevant data for better analytical performance. The types of filters include *event filters* for filtering event types (e.g., [40], [139]), *time filters* (e.g., [30], [62]) for narrowing down to a time period within the sequence, *attribute filters* (e.g., [15]) for retrieving a subset of event sequences based on sequence or event attributes, and *pattern filters* (e.g., [38], [81]) for querying event sequences that contain specific subsequences.



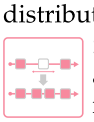
Editing enables users to modify event sequences through *adding new events*, *removing existing events*, *editing event order*, and *editing event duration*, which is commonly employed in what-if simulations of event sequence predictions. The goal is to interactively explore the influence of historical events on prediction results. For instance, in CarePre [48] and RetainVis [60], users can edit event sequences to understand how changes in individual events affect the prediction of risks.



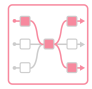
Segmentation enables users to split event sequences into sections and is typically used to narrow the scope of exploration by focusing on sequence segments that are shorter than the entire sequence. Meaningful sequence segments can also indicate event occurrence patterns. For example, in MAQUI [62] and DecisionFlow [32], users can segment a set of event sequence by user-specified milestones events to reveal event patterns and correlations.



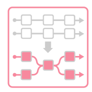
Alignment refers to arranging multiple sequences so that they are aligned based on a selected event or time point. This interaction aims to explore and compare patterns before and after the alignment point within a single sequence or across multiple sequences. For instance, Lifelines2 [115] supports the interactive alignment of event sequences based on a selected event, so that users can easily spot precursor, co-occurring, and aftereffect events. Chen et al. [15] allow both sequence alignment and adjustment of the temporal scale to illustrate the temporal distribution of events with respect to a selected event.



Scaling allows analysts to zoom in/out of visualization or inspect data under various granularities. *Zoom in/out* are commonly used to allow visualization-level scaling to enhance local details or get an overall impression. Additionally, some visual analytics techniques [15], [38], [48], [77] also allows a data-level scaling through *abstract/elaborate* to accommodate the complexity of event sequence. For example, Guo et al. [38], [39] allows a stage-level abstraction and elaboration by aggregating and expanding events within the same progression stage.



Emphasis facilitates the discovery of interesting patterns [99] through various forms of interactions such as *highlighting*, *sorting*, and *layout adjustment*. *Highlighting* draws user attention through tweaking basic visual representations (e.g., color, size), which are commonly used in emphasizing sequence groupings, progression pathways, and critical events. *Sorting* emphasizes the ranking of sequences or patterns under specific metrics. For example, Lifeflow [121] allows users to sort progression pathways by the number of records or average time span. *Layout adjustment* enables users to arrange the positions of visual elements in a meaningful way. For example, Guo et al. [40] proposed a layout algorithm that arranges sequence clusters to imply their similarities and allows users to adjust the similarity threshold to generate different groupings.



Aggregation enables users to interactively merge event sequences, supporting a more scalable exploration of large-scale, complex event sequences. For instance, DecisionFlow [32] aggregates sequences with similar occurrences of milestone events to enhance the visual scalability of large-scale events. CareFlow [79] merges sequences by common event occurrences to reveal frequently observed progression patterns.

Each visual analytics technique can be considered as a combination of design decisions in each dimension determined by the targeting analytical task and application. While the categorization of design space can help differentiate the tools in various dimensions and give an overview of their composition, we consider the analytical tasks and applications more useful to practitioners as they often have a specific analytical task and usage scenario in mind. Therefore, the main body of this survey in the following two sections (Section 4 and Section 5) is organized around the analytical tasks and applications.

4 Visual Analysis Tasks and Techniques

In this section, we introduce analytical tasks that are frequently applied to event sequence data and discuss the visual analytics techniques are designed for each task. This includes nine analytical tasks, which we further categorize into five high-level tasks: **summarization** for uncovering major progression patterns and featuring groupings of sequence entities; **prediction and recommendation** for analyzing observed event sequences to foresee upcoming events or sequences, or examining how certain interventions may effect future trends; **anomaly detection** for identifying rare cases that deviate from the majority of sequence progressions; **comparison** for investigating similarities and differences between event sequences; and **causality analysis** for uncovering causal relationships between event types to promote a better understanding of which event is likely to occur after another or of what brings about certain changes to an outcome event. As stated in Section 2.2, the analytical tasks are summarized after reviewing collected papers, which should cover most tasks for event sequences.

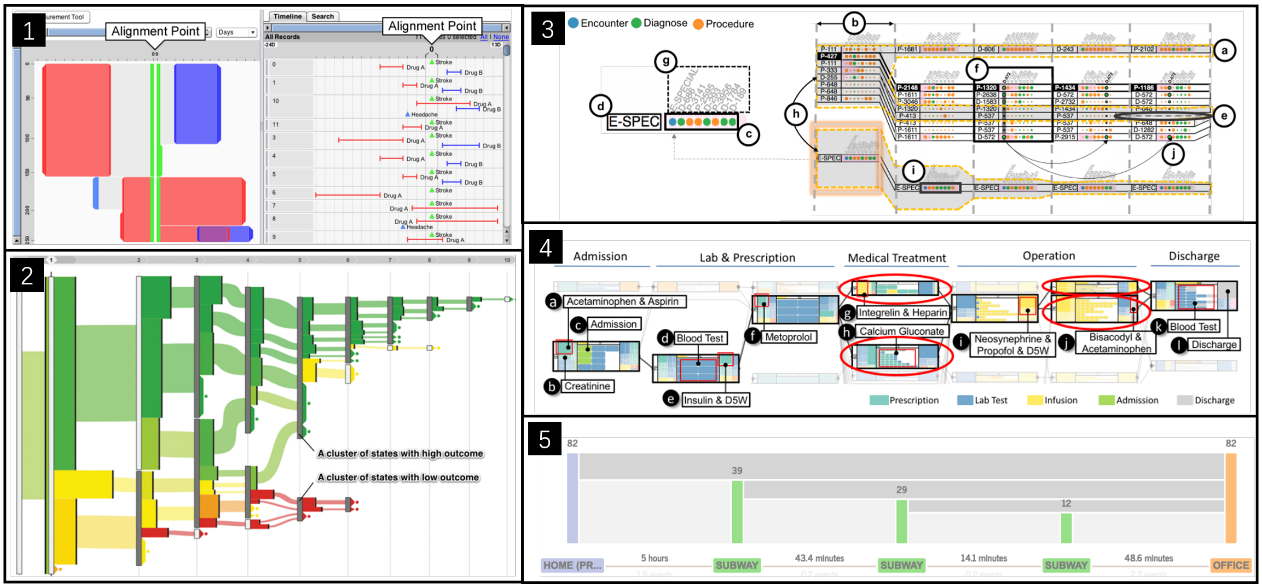


Fig. 3. Selected examples of visual summarization techniques. (1) EventFlow [72] visualizes event sequences in both an aggregated tree-like overview and detailed a timeline display. (2) Outflow [119] visualizes alternative progression paths using color-coded edges that map to patient outcome. (3) EventThread [40] visualizes the threads derived by tensor analysis as segmented linear stripes, following a line map metaphor. (4) EventThread2 [39] uses a node-link visual design to provide a higher-level summary of progression patterns of event sequences. (5) MAQUI [62] applies a hierarchy-based visualization to represent multiple frequent patterns and adopts a timeline to reveal the temporal information.

4.1 Visual Summarization

Summarization of event sequences aims to use intuitive representations to reveal major progression patterns and featured groupings of the sequence entities. In many domains such as health informatics [7], [31], [41], [78], [79], [83], [96], social media [62], [81], and career design [39], [40], a variety of analytical tasks serve the purpose of generating summaries, including **explicit summarization**, **inexplicit summarization**, **progressional analysis**, and **clustering**.

Explicit summarization techniques generate sequence groupings through aligning and aggregating common event types within the sequence. The underlying mechanism is based on iterating all permutations of the event transitions. There is a clear correspondence between the summarized patterns and the raw event sequence. Therefore, the summarized patterns are explicit and can be easily interpreted. Existing techniques adopt various visualization approaches to display event sequences, such as timeline-based [7], [54], [63], [83], [115], [135] and hierarchy-based [72], [92], [108], [121] visualizations. Timeline-based visualizations are frequently adopted to emphasize temporal ordering of events. For instances, LifeLines [83] and its variant [115] leverage timeline-based visualizations to display the temporal distribution of events in varying time granularities. Hierarchy-based visualizations, such as tree-map and icicle plot, organize the progression of events into a hierarchical structure to emphasize more on the branching of sequences. EventFlow [72] aggregates sequences into icicle plot, and display individual sequences with a list of timelines (Fig. 3(1)).

Inexplicit summarization techniques usually leverage data mining techniques to extract frequent event patterns, which only provides a partial view of the raw event sequences. Due to the loss of context, there is no explicit correspondence between the discovered pattern and raw event sequences. Existing works that serve the purpose of inexplicit summarization mainly fall into two categories: *query-based techniques* and *mining-based techniques*. *Query-based techniques* [28], [57], [111], [119], [122], [136] enable analysts to create complex queries to extract event sequences of interest. For instances, in COQUITO [57] and CAVA [136], analysts can express complex queries for iterative cohort construction. In Outflow [119], alternative clinical pathways within EMRs are visualized using a Sankey Diagram, with the color of paths representing the patient outcomes (Fig. 3(2)). *mining-based techniques* leverage advanced sequential pattern mining algorithms could extract meaningful insights from complex event sequences [61], [64], [65], [66], [81], [82]. For instances, in Frequence [81] and its variant [82], large scale EMRs data are represented by a set of extracted frequent patterns. The authors employed Sankey Diagram to reveal the correlations between treatment patterns and the associated color-coded outcomes. However, extracted patterns do not always correspond to important or meaningful information within the data. Therefore, Law et al. [62] proposed MAQUI, which interweaves *quering* and *mining-based techniques* to allow interactive querying of frequent patterns. The authors applied hierarchy-based and timeline-based visualization to represent frequent patterns and temporal information, respectively (Fig. 3 (5)). Similarly, Chen et al. [15] combines *querying* and a *mining-based* method by using Minimum Description Length Principle to extract informative patterns from event sequences with minimal information loss. Each extracted pattern is showed by timeline-based visualization.

Progression analysis aims to uncover the evolution of events during a period of time. Most of the aforementioned techniques produce highly summarized results, but fail to show important low-level event details (e.g., single event features) which can help in the crucial task of semantic interpretation of the discovered patterns [40]. Visual progression analysis techniques, such as [14], [32], [33], [39], [40], [80], have been introduced to reveal time-evolving patterns of latent progression stages. For instance, in DecisionFlow [32], analysts can use a milestone-based approach to retrieve progression patterns of interest, and visualized them in a hierarchy-based visualization. EventThread [40] has been introduced to summarize latent sequential patterns within a large-scale sequence collection. This technique employs a clustering algorithm to group the summarized patterns into various categories at different stages. A line map is employed to clearly illustrate the summarized latent patterns (Fig. 3(3)). Guo et al. further enhanced this technique with an unsupervised progressions analysis algorithm and a visual progression analysis tool, EventThread2 [39], to identify semantically meaningful progression stages. This technique solved the time scale limitation of EventThread and proposed a new visual design to demonstrate the stage-based patterns. It combines node-link visualization (Fig. 3 (4)) of sequence clusters at each stage and timeline-based visualization of individual sequences to support multi-granular analysis.

Clustering is the process of finding sequence-wide similarities to achieve sequence groupings. In the clustering analysis of event sequences, a broad range of visual analytics techniques has been developed to empower analysts working with three types of event sequence data, including temporal event sequences, spatiotemporal event sequences and microarray sequences. For temporal event sequences, clustering can be informed by sequence characteristics such as event types and sequence attributes. DICON [3] segments a collection of event sequences into groups based on entity attributes (e.g., age, gender). Each multi-dimensional cluster is revealed in a hierarchy-based visualization, which allows analysts to understand the event distributions in different groups. Cadence system [34] offers a scatter-plus-focus visualization design that supports the interactive hierarchical exploration of the space of event type groupings. This system adopts scented navigation cues to help users navigate complex hierarchies, as well as interactive bar charts and histograms that support additional constraints in categorical and continuous attributes of the target groups. [76], [93], [112], [118] are utilized to cluster individual entities (e.g., works, online users) based on behaviors. VASABI [76] summarise user behaviors by extracting their common tasks, and then identifies the groups of users based on user behaviors. This technique facilitates interactive analysis of user clustering through a hierarchy-based visualization.

The clustering analysis of spatiotemporal event sequences has been explored in many efforts, such as [45], [58], [90], [110], [131]. Spatiotemporal visual analysis of activity diary data is visualized through VISUAL-TimePAcTS [110] on a coordinate plane of time and space. Robinson et al. [90] developed STempo, a geovisualization application to facilitate the exploration of spatiotemporal patterns within event sequence data, in terms of time, geography, and content. Moreover, studies have also introduced visualization tools to cluster microarray sequences [44], [95], [98], [102]. Seo and Shneiderman [98] created the Hierarchical Clustering Explorer that offers a dendrogram and two-dimensional scattergrams, and their dynamic query controls allow users to choose which clusters to display. This model is especially suitable for bioinformatics and microarray data. Moreover, SequenceJuxtaposer [102] facilitates the comparison of biomolecular sequences using a visualization technique called "accordion drawing".

In conclusion, visual summarization techniques save user effort by capturing a broad view of event sequence data. To allow for interactive exploration of visual summarization from different perspectives, the aforementioned techniques commonly employ the following interaction techniques within their interfaces: *filter/query* for retrieving information of user interest, *scaling* for multiple scales visualization, *alignment* for aligning event sequences on selected events or time points, and *sequence editing* for modifying sequences during analysis.

4.2 Visual Event Prediction & Recommendation

There is a growing need for predictive analysis of event sequence data, which leverages historical events and making predictions about the future. It is especially useful in supporting practitioners making decisions, such as treatment plans, marketing interventions, financial investments, etc. Existing visual predictive analysis techniques can be broadly categorized into **prediction techniques**, **recommendation techniques**, and **interpretation techniques**.

Prediction techniques for event sequences have been proposed to predict the next event in a sequence based on historical events. For instance, medical researchers and physicians can use this type of technique to understand the potential outcomes of patients under different treatments. CarePre system [48] leverages attention-based RNNs to predict the risk of a patient being diagnosed with certain diseases in the future. In this system, a patient's historical events are displayed in a timeline-based visualization (Fig. 4(2)). Users are allowed to modify these events (e.g., by removing, moving, adding, or adjusting event durations) to test the impact of historical events on predicted outcomes. Guo et al. [37] also employs Recurrent Neural Networks (RNNs) to predict future activities, and review the most probable predictions and possible alternatives in a circular glyph design (Fig. 4(4)). The color of the first outer ring represents the top prediction for a group of records. Then, depending on the granularity of the analysis, alternative predictions are represented as rings and added to glyph from the inside out.

Recommendation techniques provide reliable suggestions on user actions to help achieve certain goals. For example, students can adopt this type of technique to understand their future career development and find an academic plan that suits their desired goals. Du et al. [22], [23] introduced two career path recommendation techniques that provide suggestions and potential outcomes by summarizing the outcome of similar users. In EventAction [22], all of the records that similar to the target record are displayed in a list of calendar views (Fig. 4(1)). Recommended actions are highlighted in the calendar with green and allow users to add into their plans for exploration.

In the past few years, deep learning algorithms have demonstrated significant improvements over traditional approaches in predictive analysis. For event sequence data, Recurrent Neural Networks (RNNs) are frequently adopted to foresee the upcoming events or sequences, or exam how certain interventions may affect the future trends. However, interpretability is recognized as a primary challenge of deep learning approaches. To address this issue, recent studies have introduced **interpretation techniques** in visual predictive analysis to interpret the internal mechanisms of a prediction model [56], [60], [71], [104]. For instance, RetainVis [60] is a hybrid visual technique for gaining insight into how RNNs model EMR data within the context of diagnosis risk prediction tasks. This technique interprets the relationship between patient records and predicted risk scores. Specifically, patients' medical records and their predicted risk trajectory are visualized in two parallel line charts

Fig. 4. Selected examples of visual prediction & recommendation techniques. (1) EventAction [22] uses a calendar view to show the temporal information of event sequences. (2) CarePre [48] reveals the medical record of a patient in a timeline-based visualization, and similar patients' medical event sequences are aggregated into a Sankey-based visualization. (3) In RetainVIS [60], predicted risk trajectories are revealed in parallel line charts (middle), and the risk contributors for the patients are displayed in a bar chart (bottom). (4) [37] visualizes the top prediction, alternatives, and their uncertainties in a circular glyph design.

(Fig. 4(3)), which allow users to understand the progression of predicted diagnosis risks and why such predictions are made. Also, when users hover over the x-axis, they can see the updated contribution scores of medical events, which represent the importance or contribution of an event to the predicted result. Similarly, LSTMVis [104] focus on the visual analysis of hidden features in RNNs, it allows users to explore hypotheses about hidden state dynamics.

Visual prediction and recommendation techniques contribute to decision-making in many domains. In order to allow users to explore the data from different perspectives, the aforementioned techniques commonly employ the following interaction techniques within their interfaces: *filter/query* for retrieving information of users interest, *emphasis* for adjusting attributes of data to reveal interesting patterns, and *sequence editing* for including a new event or a new feature into the prediction model.

4.3 Visual Anomaly Detection

Visual anomaly detection for event sequences aims at identifying rare cases that deviate from the majority of the sequence progressions. It has been applied in many applications, such as social media [4], [6], [138], computer systems [73], [97], [129], clickstream [30], [38], [77], and smart factory [43], [123], [130]. As the forms of anomalies vary from task to task, existing visualizations for anomalies in event sequence data can be categorized by data scales, including **anomalous events visualization**, **anomalous frequent patterns visualization**, and **anomalous sequences visualization**.

Anomalous events visualization aims to distinguish anomalous events from normal events within the context of event sequences. Existing techniques leverage various visualization methods to display anomaly events [8], [30], [38], [74], [77], [130]. For instance, EventThread3 [38] detects redundant and missing events within anomalous sequences by comparing anomalous sequences with the inferred expected normal progressions (Fig. 5(4)). The anomalous events are displayed in a line of circular glyphs with the size encoding the level of abnormality, which can help analysts understand why the sequence is identified as an anomaly. Xu et al. [130] extended Marey's graph to reveal the working status of a production line. This design visualizes the moving traces and processing times of individual products, improving users' understanding of the overall performance of the production line, the anomalous events, and the causes and effects of the anomalies.

Anomalous frequent patterns visualization is utilized to help users perceive the anomalous frequent patterns that contribute to sequence abnormality. MOOCad [73] is designed to detect anomalous learning patterns within MOOC data (a set of online learning activities sequences) (Fig. 5(2)). To facilitate anomaly detection and reasoning, the large-scale learning sequences are clustered into various groups at different stages. The authors employed a Sankey-based visualization to aggregate the stage segmentation results. A matrix-based visualization is also employed to indicate the behavioral patterns of each student group within the stage, which facilitates the comparison of patterns across stages, between groups and individual paths.

Anomalous sequences visualization helps users detect anomalous sequences within a collection of sequences, and uncover the deviation of anomalous sequences from normal sequences.For instance, Zhao et al. [138] proposed a flexible timeline visualization to discover rumor-spreading processes between Twitter users (Fig. 5(3)). The retweeting sequences are visualized with a design of packed circles, each representing a participating user. In order to intuitively display the abnormality of sequences, the authors designed a circular glyph for each retweeting sequence which summarizes important metrics such as overall abnormality, contextual polarity, scale, and temporal information. Similarly, Cao et al. developed TargetVue [6] to analyze anomalous behaviors of Twitter users. This technique analyzes sequences of retweeting behaviors and identifies anomalous users via an unsupervised learning model. The behaviors

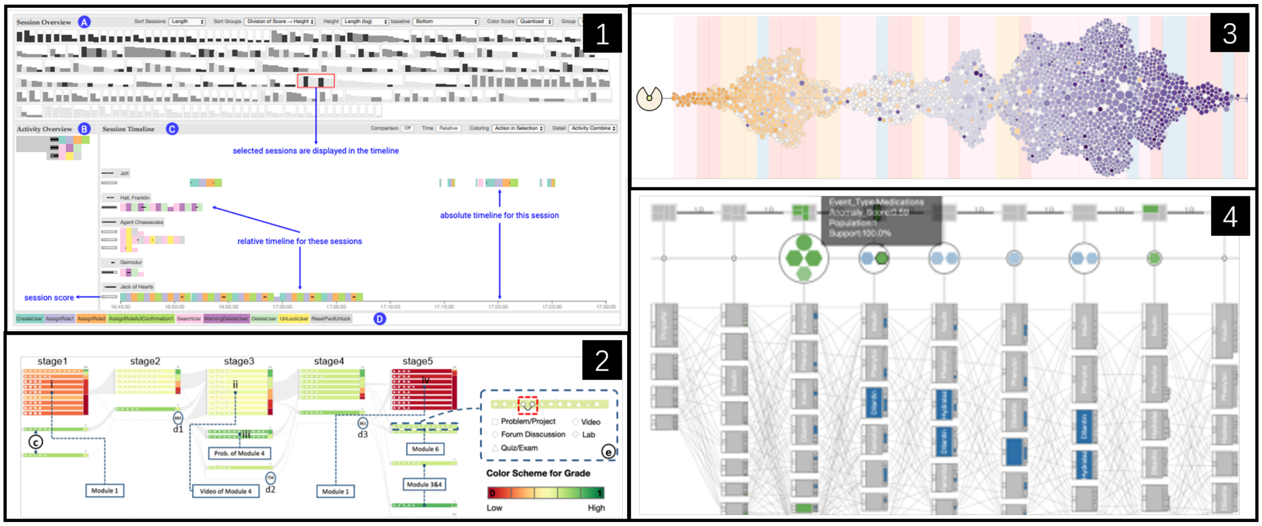


Fig. 5. Selected examples of visual anomaly detection techniques. (1) [77] combines a rectangle glyph design and a timeline-based visualization to reveal anomalies within event sequences. (2) MOOCad [73] employs a Sankey-based visualization to display an overview of the stage segmentation results, and it uses a matrix-based approach to indicate the content patterns of each group within the stage. (3) FluxFlow [138] visualizes anomalous retweeting sequences in a packed circles design. (4) EventThread3 [38] displays anomalous sequences in a line of rectangular nodes ordered by time of occurrence (top). The authors use a circular glyph (middle) to visualize the anomalous events within sequences.

of suspicious users are summarized by three glyph designs (Fig. 9(4)), presenting the users' communication activities, features, and social interactions, respectively. Nguyen et al. [77] proposed a visual analytics approach that helps identify and analyze unusual action sequences (Fig. 5(1)). Actions in each sequence are visually summarized in a compact glyph to help analysts find anomalous sequences, with the length and color saturation of glyph representing sequence length and anomaly scores, respectively. Each individual anomalous sequence is also visualized in a timeline visualization, where each event is represented by a rectangle colored by event type. Similar designs are also employed in [30] and [38]. Guo et al. [38] further employed MDS projection of the sequences in the dataset to give an overview of the distribution of anomalous sequences.

To allow users to interactively explore data from different perspectives, the aforementioned techniques commonly employ the following interaction techniques: *filter/query* for retrieving information of users interest, *emphasis* for adjusting attributes of data to reveal interesting patterns, *scaling* for multiple scales visualization, and *alignment* for aligning sequences on selected events or time points.

4.4 Visual Comparison

Visual comparison is for investigating the similarities and differences between event sequence data. A variety of visual comparison techniques have been proposed to solve real-world problems in many domains such as career path planning [22], [24], clickstream analysis [77], [139], health informatics [36], [48], [68], and general event sequence comparison [38], [122]. We classify prior visual comparison techniques for event sequences based on the granularity of compared targets, including **comparison of individual event sequences**, **comparison of sequential patterns**, and **comparison of sequence collections**.

Comparison of individual event sequences aims to identify disorder, missing or redundant events, as well as differences in event attributes (e.g., timestamp, event duration, etc.) in a target sequence compared to a base sequence. To facilitate interpretation of compared results, researchers adopted *juxtaposition design* [122], *superposition design* [48], and *hybrid design* [36], [38] to clearly visualize the similarities and differences between sequences. For instance, Similan [122] shows the similarity of events within two similar sequences via *juxtaposition design*. Each event sequence is visualized in a binned timeline (Fig. 6(4)), with the base sequence placed beneath the target sequence for explicit comparison. In CarePre [48], *superposition design* are utilized to compare the predicted medical record with records of similar pre-existing patients. In the most recently published visual comparison technique, Guo et al. [36] searched similar medical records of a target sequence and applied *hybrid design* to convey the differences between the target record and its similar records. This technique uses *explicit encoding* to display the dissimilarity over time, and it uses *superposition design* to support analysts manually compare the target sequence with similar records sequences.

Comparison of sequential patterns aims to investigate the similarity of sequential patterns within two event sequences. Typical applications include comparing frequent patterns in log files [77], [87], and understanding behavioral patterns in career paths [22], [24]. For example, EventAction [22] compares the sequential pattern between students to retrieve students that are similar to the target student. It further leverages a *juxtaposed* calendar view to compare events in the sequences. Nguyen et al. [77] adopted a *superposition design* to display sequential patterns of the abnormal sequences. Du et al. [24] support both *explicit encoding* and *juxtaposition* to demonstrate the differences of similar sequences (Fig. 6(1)). Specifically, when comparing the target record with the entire dataset, Du et al. summarized the metrics for determining the similar records in a hierarchical tree, where the similarities and differences are explicitly encoded. For a detailed inspection, all records and common temporal patterns are visualized in the calendar views, so that users can juxtapose any two sequences of interest to investigate the similarities and differences between them.

Comparison of sequence collections aims to find differences between two sets of event sequences in terms of sequence structure, event attribute, and temporal information [67]. For instance, CoCo [68] leveraged statistical analysis

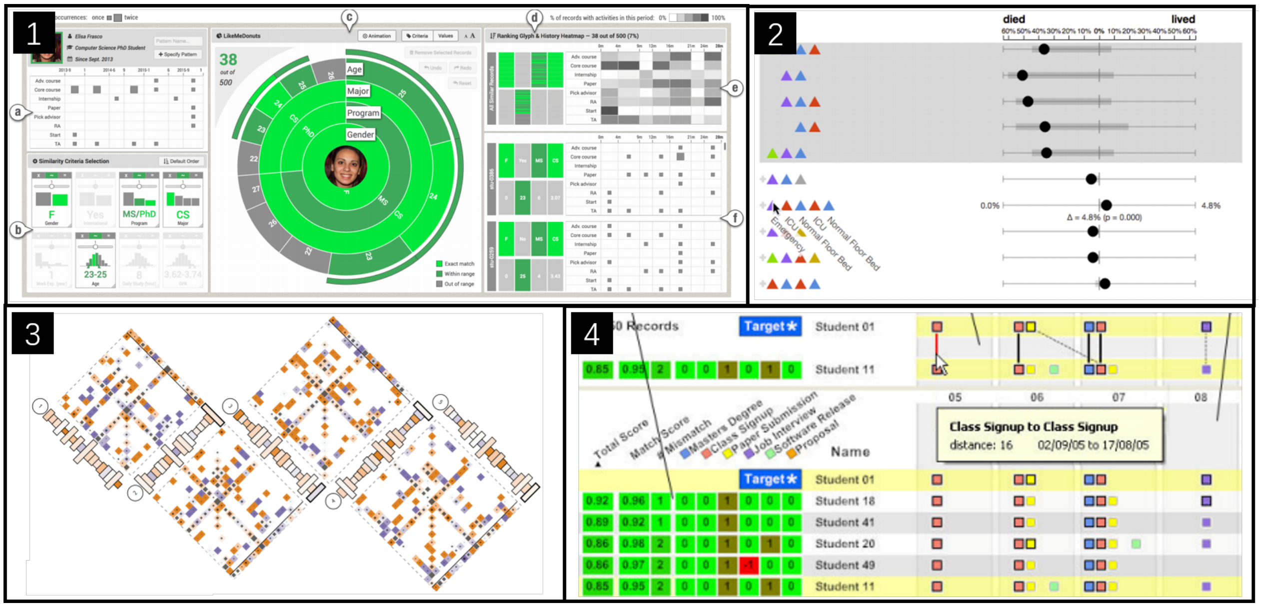


Fig. 6. Selected examples of visual comparison techniques. (1) [24] summarizes the criteria values of similar records as a hierarchical tree. Each of the records and common temporal patterns among the similar records are visualized in the calendar view. (2) In CoCo [68], a combination of medical events is visualized in a timeline, where colored triangles represent medical events. (3) MatrixWave [139] visually compares two web clickstreams in a matrix-based visualization. (4) Similan [122] visualizes each event sequence in a binned timeline. The paired events are connected by lines.

to compare the attributes of two distinctly defined cohorts, adopting *explicit encoding* to convey an overview of the differences between the two cohorts (Fig. 6(2)). MatrixWave [139] utilized a novel matrix-based visualization to compare the traffic of two web clickstream datasets (Fig. 6(3)). It applied *superposition design* to display two event sequence datasets in one visualization and used *explicit encoding* to show the differences in the amount of traffic between pairwise steps of two clickstream datasets.

In summary, visual comparison techniques can save analysts' efforts to explore the differences between two event sequences or two groups of event sequences. To facilitate interactive analysis, the aforementioned techniques adopt *filter/query* for retrieving information of users interest, *scaling* for multiple scales visualization, *alignment* for aligning sequences on selected events or time points, and *emphasis* for adjusting attributes of data to reveal interesting patterns.

4.5 Visual Causality Analysis

The problem of learning causality in data has attracted much attention in data mining community over the past years. To facilitate interactive exploration and help with better understanding of complex causal relations, many visualizations and visual analytics techniques have been introduced to support causality analysis. However, most prior researches focus on analyzing causalities in non-temporal multivariate data, while the visual causality analysis for event sequence data is still under explored. In this section, we review existing visual causality analysis techniques for multivariate data and event sequence data. Traditional visualization for displaying causality includes Directed Acyclic Graph (DAG) and Hasse diagram [55], which are commonly applied and easy to understand. However, when the number of variables increases, traditional visualizations may suffer from the exponential growth of edge crossing. To address this issue, Elmqvist et al. proposed Growing Squares [26] and Growing Polygons [27], which substitute nodes in DAG with color-coded squares and polygons to provide an overview of influences on each event in place of the links. Furthermore, they incorporate animations to present the temporal ordering of causality. Kabada et al. [51] introduced a set of animations following Michotte's rules of causal perception [70] to imply causal semantics, such as causal strength, amplification, dampening, and multiplicity in causality visualization.

To further facilitate interactive causality analysis and interpretation, recent studies attempt to develop visual causality analysis systems that incorporate automatic causality analysis algorithms with causality visualizations. Chen et al. [9] proposed a workflow for visual causal analysis, which leads to a number of visual analytics systems that are designed to support interactive analysis and reasoning of causation. For example, Zhang et al. [137] introduced a visualization tool that analyzes causality between numerical and categorical variables in multivariate data and utilized force-directed graphs to display the causality. ReactionFlow [20] uncovers the causal relationships between proteins and biochemical reactions in biological pathways. The causal pathways are organized into a Sankey-based structure to emphasize the downstream and upstream nature of the causal relationships. Want et al. [113] presented a visual analytics tool that allows analysts to edit the causality analysis result according to their domain expertise. They further enhanced this techniqiue in [114] with a path diagram visualization to better expose the causal sequences.

As prior efforts mainly focus on the causal analysis of multivariate data where no temporal information is involved, a few most recent techniques have been proposed to analyze causal relationships among events in event sequence datasets [18], [49], [127]. Causality Explorer [127] enables users to explore, validate, and apply causal relations in high-dimensional event sequence data. The tool provides an uncertainty-aware causal graph visualization to present a large set of causal relations inferred from event sequences. SeqCausal [49] recovers the Granger causality of events within a collection of event sequences based on Hawkes process modeling, it leverages a set of visualizations and inter-

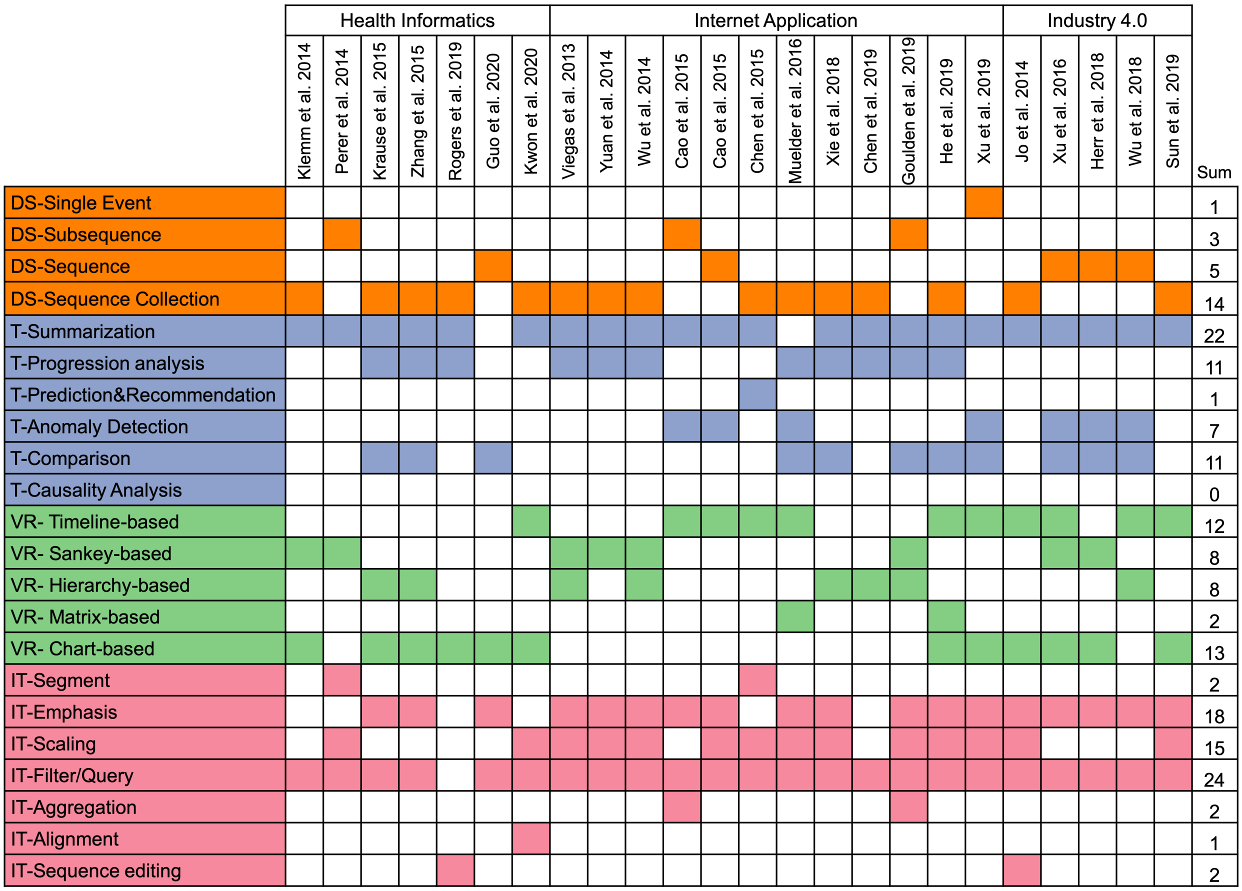


Fig. 7. The selected papers regarding visualization and visual analytics of event sequences in different application domains. Each paper is labeled by relevant analysis tasks and design components in the design space. The rows are grouped and colored by dimensions of our proposed design space: DSs - Data Scales; Ts - Analysis Tasks; VRs - Visualization Representations; ITs - Interaction Techniques.

actions to explore, interpret and verify complex causalities in high-dimensional and heterogeneous event sequences.

5 Applications

In this section, we introduce application areas where event sequence data are commonly observed and discuss the analytical challenges that are specific to each application domain. This includes five applications, which we further classify into three high-level fields: **health informatics** for electronic medical records; **internet applications** for general behavioral data collected from websites and computer systems, and **industry 4.0** for manufacturing data. As stated in Section 2.2, the applications are gathered after a comprehensive paper search in major data mining and visualization journals and conferences, which should cover most event sequence applications. While data from other application domains may also be easily transformed to the form of event sequence data (e.g., segmenting continuous time-series data and traversing graph data to generate discrete event sequences), they are out of the scope of this paper. In this section, we set our focus on applications only with data that are naturally perceived as event sequences.

5.1 Health Informatics

Electronic health records (EHRs) or electronic medical records (EMRs) represent a typical form of event sequence data. The record of each patient over the course of a clinical process can be considered as an event sequence, with each event representing a clinical event, such as a diagnostics event, taking lab tests, doing surgery or taking medicine. With ample medical event sequence data and domain knowledge, physicians and medical researchers can extract new knowledge, quantify the effects of changes in care delivery, and potentially guide the formation of best practice guidelines. For example, medical experts are often interested in identifying the commonness in clinical pathways among different patients, so as to extract universal clinical guidelines that are applicable to cohorts with certain characteristics(e.g., treatment customization). Besides analyzing historical data, predictive analytics are also found to be useful in medical applications, such as predicting disease outcomes and complications, so as to help doctors adjust their treatment plans beforehand. However, EHRs can be difficult to analyze due to the numerous set of unique event types and the subsequent heterogeneous sequence progressions. EHRs of patients diagnosed with chronic diseases also suffers from large time span and sequence length. A variety of visual analytics solutions have been proposed to address these challenges, which we summarize in the following based on analytical tasks,including **cohort analysis** [3], [39], [40], [57], [59], [67], [136], **outcome analysis** [32], [34], [79], [91], [116], [119], [121], and **progonsis analysis** [48], [60].

Cohort analysis is a common approach used to uncover correlations between a specific disease risk and the underlying attributes of patients within the cohorts [136]. Medical researchers can construct a cohort of patients based on a medical event (e.g., diagnosis, treatment), the attributes of patients (e.g., gender, age), and the patterns of individual sequences (e.g., symptoms progression, treatment progres-

Fig. 8. Selected examples of visual techniques for health informatics. (1) COQUITO [57] uses a hierarchical tree map and bar charts to provide an overview of statistical information of cohorts defined by users. (2) Composer [91] plots the outcome score trajectories of different procedures in a line chart. (3) CAVA [136] uses a stacked bar chart, a pie chart, and a hierarchical tree map to represent the age, gender, and diagnosis distributions of a cohort, respectively. Both the calculated risk scores and event progressions within the cohort are visualized by color-coded edges.

sion). Suppose a medical researcher wishes to understand the exposure factors for lung cancer. He can gather the answer by analyzing common attributes within a cohort of lung cancer patients or by measuring the differences between cohorts with or without lung cancer. Following this idea, existing visual techniques for cohort analysis emphasize one of two strategies: cohort summarization [39], [40], [86], [136], or cohort comparison [3], [57], [59], [67], [85].

Cohort summarization techniques, such as CAVA [136] and Chronodes [86], visually summarizing informative patterns within a cohort and uncover the common exposure factors for a disease. CAVA [136] combines chart-based and hierarchy-based visualization to represent the attribute distributions of a cohort (Fig. 8(3)). Then, to further investigate exposure events in the cohort, each patient was assigned a hospitalization risk score based on their medical history. Both the calculated risk scores and event progressions are visualized by color-coded edges, analysts can intuitively understand how different event progression pathways lead to different hospitalization risk scores and which medical events have higher risks. Moreover, in EventThread2 [39], the clustered medical event sequences and common sequential patterns (e.g., typical care plans) of a cohort are visualized in a Sankey-based visualization and a node-link visual design respectively (Fig. 3(4)). User can inspect common sequential patterns of a cohort with the goal to explore those medical events that affect further progression.

Cohort comparison measures differences between two cohorts of patients to determine exposure factors of a condition such as a disease or death. COQUITO [57] helps users interactively construct two cohorts and explore exposure events for a disease. It uses a hierarchical treemap and multiple bar charts to provide an overview of statistical information about the cohorts (Fig. 8(1)). Then it leverages PARAMO [75] to compare two cohorts and determine if the constructed cohorts carry exposure events for a disease. CoCo [67] is a visual comparison technique (Fig. 6(2)) that measures the differences between two cohorts under various differentiating metrics. Users can select metrics of interest, such as the most differentiating event subsequences between two cohorts, to explore the medical events or patterns that may influence the incidence of a condition. In CoCo, each row displays the difference between two cohorts and the medical patterns of cohorts are visualized by a timeline-based design. A circle marker is placed horizontally between two cohorts to display the difference between the values and in the direction of whichever cohort's value (e.g., death rate, survive rate) is higher.

Outcome analysis studies the end results of different medical progressions (e.g., symptom progressions, treatment progressions) with the goal of facilitating informed decision-making about diagnosis and treatment options. Existing works, such as Outflow [119] and Frequence [81], reveal the medical progression paths in a Sankey-based visualization to uncover the outcomes of different procedures. More specifically, Outflow [119] aggregates medical event sequences from a cohort of patients and visualizes alternative progression paths using color-coded edges that map to patient outcomes (Fig. 3(2)). Similarly, in a series of efforts proposed by Perer et al. [79], [81], [82], the authors extracted frequent progression pathways of a cohort and used a Sankey-based visualization to display them while providing context on which care plans were successful and which were not. These techniques provide an overview of the progression pathways within a cohort, and thus help users understand which factors, medical pathways, or other structures are most associated to the outcome of interest. Nevertheless, as users are not allowed to interactively build the cohorts in some outcome analysis techniques, the analytic capability of these techniques could be hugely impacted when analyzing a sequence collection of different patients. To overcome this issue, DecisionFlow [32] leverages a milestone approach to support users in defining a cohort by highlighting patients with a specific outcome (e.g., a disease). Composer [91] enables users to interactively explore the outcomes under different cohorts and treatment plans. This technique employs PROMIS (Patient-Reported Outcomes Measurement Information System) to automatically evaluate the outcome scores of a patient under user-defined treatments, and plots the outcome score trajectories in a line chart (Fig. 8(2)). Medical researchers can plot outcome trajectories of different treatments in one chart to determine the optimal treatment for a cohort of patients.

Prognosis analysis predicts the risks of a patient being diagnosed with certain diseases in the future based on the patient's medical history. A series of deep learning–based visual prognosis techniques, such as [16], [48], [56], [60], have been introduced to make prognosis analysis and interpret the results. For instance, [56], [60] implement RNNs to predict the current and future states of a patient. RetainVis [60] enables users to modify individual sequences of medical events (e,g., add or remove medical events, modify visit period) to experiment with how predicted risk changes with the historical record. The predicted trajectory and historical medical event sequences are visualized in two parallel line charts (Fig. 4(3)). Users are able to observe correlations between medical event sequences and prediction risks, and understand why such predictions are made. CarePre system [48] can also predict the risk of a patient being diagnosed with a certain disease and estimates the most influential treatments for a patient based on historical medical records. The patient's historical events are visualized in a timeline-based visualization (Fig. 4(2)), and users are allowed to modify these events (e.g., removing, moving, duration adjustment, adding) to test with different predictions. Clinicians can create multiple edited sequences to compare the predicted results under alternative treatments, so as to understand the impact of different treatment options.

5.2 Internet applications

In various internet applications, the activities of users and devices can be recorded as individual event sequences. For instance, social media data contain sequences of timestamped activities (e.g., posting or commenting) for specific users that are recorded over time. Similarly, clickstream data collected from e-commerce websites record how visitors operate and navigate through a website, and this data can be represented as sequences of timestamped events (e.g., visiting a product page, purchasing a product) generated by visitor actions. Additionally, the system logs collected from a computer system can also be represented as temporal event sequences of device conditions (e.g., usage, temperature, workload). In this section, we provide a review of the visual techniques that have been developed for event sequence data retrieved from **social media platforms**, **e-commerce websites**, and **computer systems**.

5.2.1 Social media

On social media platforms such as Twitter and Facebook, user activities can be recorded as event sequences. Each sequence may record the temporal activities of a user over time, where each event represents an online activity such as posting or commenting. Analyzing event sequences on social media platforms could help sociologists understand the underlying behavioral patterns behind the spread of information. For example, the spread of a rumor may start with increased reposting interactions with an influential user. Such users can be identified through detecting anomalous sequences to stop the spread of misleading information at an early stage. In contrast to EHRs, sequences collected from social media platforms generally contain a limited number of event types (i.e., user interactions). However, the contextual information of each event, such as the content of a post and the underlying social network among users adds additional complexity to the data and also needs to be considered when analyzing behavioral patterns. Moreover, the large scale of users in social media platforms also leads to great difficulty in sequence analysis. Existing efforts have proposed a range of visual analytics techniques to help yield insights about various types of user behavior, including **collective behaviors** [13], [62], [81], [109], [125], [132], [138] and **ego-centric behaviors** [4], [6], [12].

Collective behaviors refer to activities conducted by a temporary and unstructured group of people. On social media, collective behaviors are formulated by groups of social media users through the processes of spreading information and human mobility. To study these collective behaviors and identify behavioral patterns, various visual analytics techniques have been proposed: [5], [13], [106], [109], [138] analyze the behavior of *spreading information*, and [62], [81] are developed to study *human mobility*. *Reposting process* refers to how information spreads across space and time on social media platforms. Google + [109] interweaves node-link diagrams and circular map metaphors to visualize message spreading paths. Analysts can easily capture the traces of diffusion between users and identify the importance of a message by its size and diffusion path. Chen et al. [13] used a map metaphor to symbolize the reposting process in a spatial context (Fig. 9(1)). The diffusion structure is visualized using various link metaphors such as rivers, routes, and bridges. This technique highlights the influence of key players, and enables analysts to explore how key players promote the evolution of topics and enlarge the influence of the source message. Zhao et al. [138] proposed a flexible timeline visualization to reveal the rumor-spreading process among Twitter users. Some studies trace the spatiotemporal information of diffusion pathways to uncover how information is spread on a global scale [5], [106]. For example, Cao et al. [5] summarized the temporal trends, the social-spatial extent, and community response to a topic with a sunflower metaphor. The original tweets are placed at the center of the circle and linked with geo-groups (users from the same country) once users in these groups repost the original tweets. The retweeting activities are displayed as a sequence of color-coded glyphs moving along pathways that indicate the timing and sentiments of the retweets. Besides the reposting process, another important collective behavior is *human mobility*. The spatiotemporal event sequences retrieved from social media platform, like Foursquare, have recently been used to uncover user mobility patterns and predict mobility decisions. For example, advertisement companies can investigate the mobility patterns of people, such as when and where they go to work, to optimize their advertising strategies. Some visual analytics techniques that leverage pattern mining algorithms have been used to explore common mobility patterns of users, such as [62], [81]. MAQUI [62] support the interactive exploration of data collected from Foursquare to

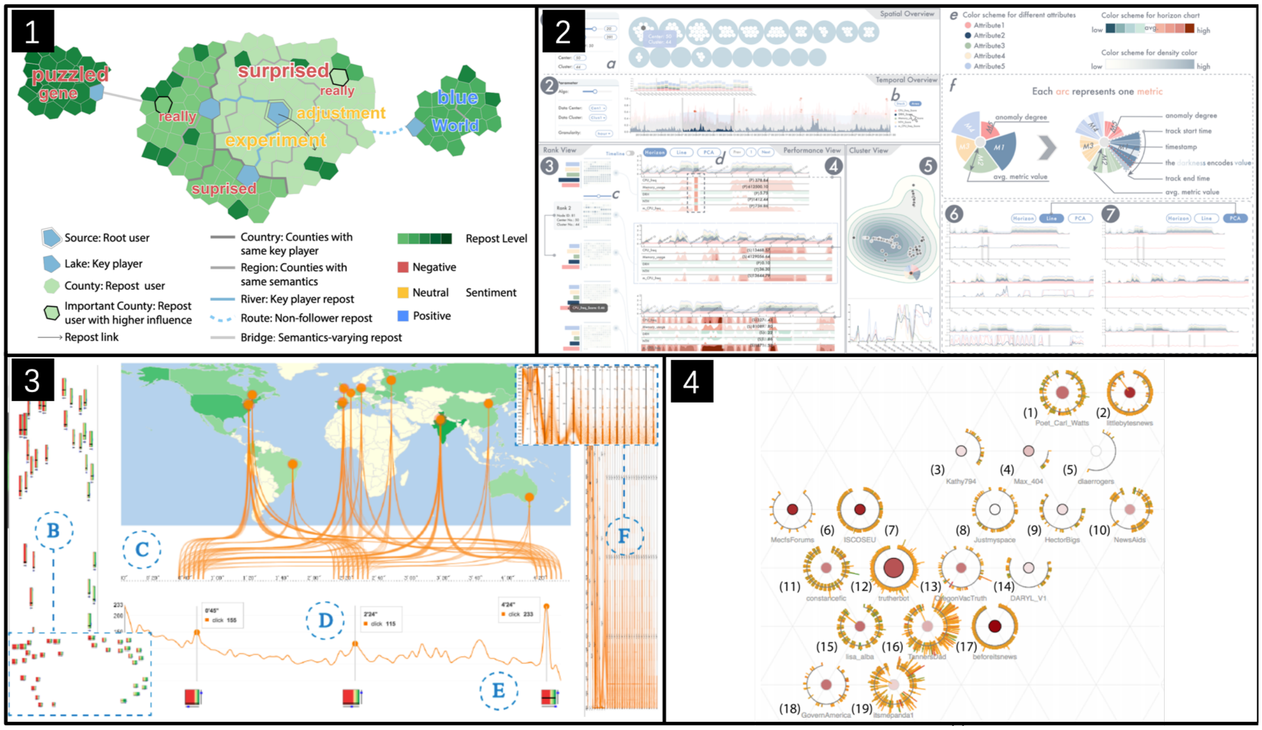


Fig. 9. Selected examples of visual techniques for Internet applications. (1) R-Map [13] uses a map metaphor to symbolize the reposting process in a spatial context. (2) CloudDet [129] combines a glyph design and a stacked line chart to monitor the performance of a computer system. (3) PeakVizor [10] encodes each interaction peak by a glyph in an overview, and the spatial-temporal information of the peaks and the correlation between the peaks are visualized in two additional views. (4) TargetVue [6] visualizes the behaviors of suspicious users in three glyphs that present the user's communication activities, features, and social interactions respectively.

uncover the frequent mobility patterns of users.

Egocentric behaviors refer to activities conducted or influenced by a user. An egocentric perspective enables a closer analysis of individual behaviors and thus provides more detailed behavioral patterns [12]. For instance, Cao et al. [4] proposed Episogram, an egocentric representation for visualizing individuals' interaction histories (e.g., posting or reposting content). Episogram visualizes each interaction thread using a vertical line on a timeline and uses a glyph design to represent interaction events among users. Building upon preceding works, Cao et al. developed TargetVue [6] to detect and visualize users with anomalous behaviors on Twitter. TargetVue detects anomalous users via an unsupervised learning model and visualizes the behaviors of suspicious users in three glyphs representing the communication activities, features, and social interactions, respectively (Fig. 9(4)). Chen et al. [12] proposed a map-based visual technique to summarize the historical diffusion traces initiated by a central user. Users who participated in reposting one central user's post are visualized as hex nodes whose color and size encode the user's behaviors and roles. These users are grouped into different regions on the map and linked with the egocentric user in the form of a social network, helping analysts trace how information reaches and diffuses from the user.

5.2.2 E-commerce

Clickstream data collected from e-commerce websites record how visitors operate and navigate through websites. A visitor's online activity can be recorded as an event sequence, in which each event represents a single online activity (e.g., visiting a product page). The increasing availability of such event sequence data permits analysts to extract valuable insights into website design and commercial activities. For instance, these meaningful insights can help companies plan efficient marketing strategies and make more precise advertising and promotion to customers. Generally, online activities on different web pages are considered as different event types, so that an event sequence of a web visitor can be considered as a description of how the user navigates through web pages. When analyzing activities in large web applications, the main difficulty usually lies in the large space of web links that users need to explore. The diverse types of online activities (e.g., click, scroll, mouse move) further increase the complexity of the data. Existing visual techniques have been introduced to explore **frequent visiting traces** [65], [66], [133] and **user behavior patterns** [10], [29], [35], [42], [73].

To facilitate the understanding of frequent visiting traces, Zgraggen et al. [133] proposed (s|qu)eries to visualizes regular patterns of clickstream data. Moreover, Liu et al. [66] extracted frequent browsing paths from clickstream data and visualized them in a funnel-based visualization. As frequent patterns do not always correspond to important or meaningful information within data, CoreFlow [65] leveraged a tree-based visualization to facilitate branching pattern exploration for browsing paths.

Analyzing clickstream data can help e-commerce companies explore user's behavior and optimize their business plans. This idea has been extended to online education platforms to analyze the learning behaviors of students [10], [11], [35], [42], [73]. For instance, PeakVizor [10] analyzes students' interaction activities to understand how students respond to video materials. For example, an unexpectedly high occurrence of pausing or rewinding may indicate that students having difficulty in understanding the material. The authors encoded high pausing or rewinding activities

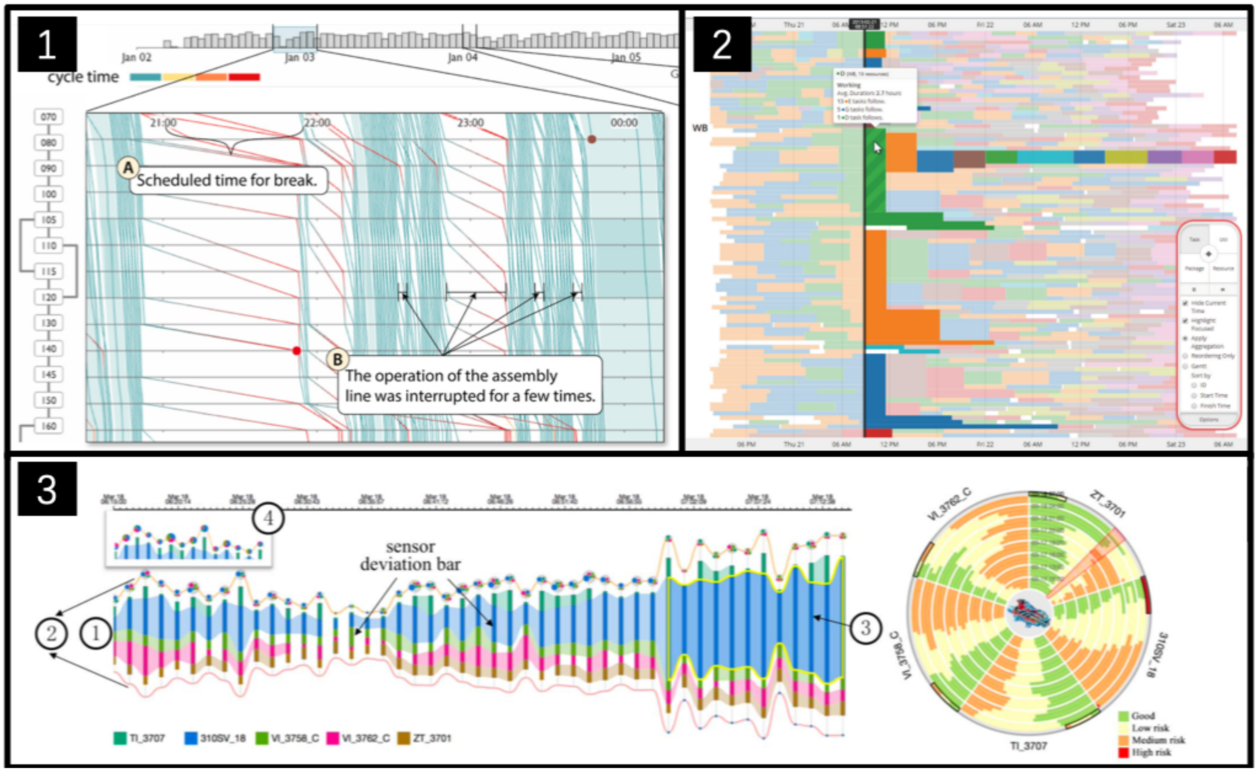


Fig. 10. Selected examples of visual techniques for Industry 4.0. (1) Xu et al. [130] extend a Marey's graph to visualize product moving traces in a production line. (2) In LiveGantt [50], the big picture of the current schedule is visualized in a Gantt chart. (3) Wu et al. [123] employ a stacked timeline to reveal how the real equipment conditions deviate from "normal" conditions in a period of time. The long-term trends of equipment conditions are visualized as a radial visualization to provide users with an overview of equipment conditions during a certain past time period.

with peaks, and provide statistics such as peak duration, and user distribution with superposed glyphs. Moreover, spatiotemporal information the correlations of the peaks are visualized with two additional visualizations (Fig. 9(3)). CCVis [35] explores the patterns of students' clicking behavior and identifies course resources that were clicked most and least. It visualizes the critical sequences that lead to different transition probabilities in a node-link diagram with a Sankey-based visualization showing the click patterns.

5.2.3 Computer systems

Computer systems are monitored by regularly sampling profile data that record the timestamped conditions (e.g., CPU load, memory usage) of specific devices over time as event sequences. Monitoring and analyzing the profile data is an efficient approach to observe the running status of a large number of devices. For instance, a cloud computing system is normally comprised of thousands of parallel computing and storage devices. Tracing every device is untenable due to scale. By analyzing the profile data, the devices that are over- or under-allocated, inefficient operations, and nodes that are misbehaving or failing could be detected from all the devices. However, the profile data collected from computing systems are large-scale and multidimensional. In order to effectively observe the data, various techniques have been introduced to overcome this challenge. Muelder et al. [74] proposed a visual technique to portray the behavior of cloud computer systems over time. The authors adopted a stacked graph timeline to summarize the aggregate behavior of cloud computing systems. For detailed inspection, the behavioral lines of each compute node are plotted in a table of line charts. In this view, analysts can efficiently explore the trends and anomalies within a system. Xie et al. [126] leveraged one-class support vector machines to detect anomalous executions in high performance computing clusters. Detected anomalies are visualized in a multi-level visualization system for deeper analysis. Specifically, all of the anomalous compute trees are identified in a scatter plot. Analysts can select the anomalies of interest to inspect their structural patterns in a node-link diagram and their invoked functions in a stacked timeline. [129] provides interactive visualization capabilities that enable analysts to inspect profile data and identify anomalous performances in cloud computing systems. This system combines multiple visualization modes, such as glyph design and stacked line charts, to monitor the performance of cloud computing systems from different aspects (Fig. 9(2)).

5.3 Industry 4.0

Industry 4.0 or The Fourth Industrial Revolution is the ongoing process of using modern smart technology in industrial practices to achieve the automation of traditional manufacturing. The temporal status of smart equipment over time can be recorded as an event sequence, where each event represents a status (e.g., an equipment condition or a processing event). The increasing availability of such event sequence data permits manufacturing experts to better understand the line's performance and explore ideas for improvement. For instance, experts could identify the recurring error patterns from collected data to indicate some systematic production issues. However, the data collected from smart factories are multivariate and high-dimensional, it is often unclear, which subsets of the data should be focused on to detect anomalies and improve the factory's productivity. To address these challenges, a variety of visual techniques have been introduced to help users **exploring**

anomalous events [8], [43], [123], [130], [141] and **optimizing manufacturing plans** [50], [105].

In smart factories, an anomalous event (e.g., equipment failure, outlier process) could result in a serious incident or great financial loss. Traditional anomaly detection depends on manually checking every equipment, which is too expensive and inefficient. In contrast, the collected manufacturing data provides a more reliable resource for factory managers to analyze anomalies. For instance, Herr et al. [43] analyze event reports of a production line and detected systematic issues in manufacturing processes. Reported events are shown as a time series plot that can help understand the error distribution and recurring error patterns. Xu et al. [130] extended Marey's graph to visualize product moving traces in a production line (Fig. 10(1)). The visualization of individual products and their processing times improves user understanding of a line's performance, and also helps in better understanding anomalies, the causes and the effects in a production line. The visual technique proposed by Wu et al. [123] provides an interactive interface to monitor the status of equipment in smart factories. The authors estimated normal conditions of equipment based on a training set, and then employed a stacked timeline to reveal how the real equipment data deviate from estimated normal conditions over a short period of time (Fig. 10(3)). Moreover, in order to visually summarize the long-term trends of equipment conditions, the authors adopted a radial visualization to provide an overview of equipment conditions during a certain past time period.

Analyzing manufacturing data can help managers and factory planners optimize manufacturing schedules. More specifically, in a production line, each machine is responsible only for a specific part of the production process. When the cooperation of machines is not well designed, the production line's overall efficiency will be negatively affected. The event sequence data of production lines record the past and current tasks of machines. By analyzing these data, factory planners can explore and reschedule inefficient plans, such as a manufacturing plan with significant equipment conflict. LiveGantt [50] is an interactive schedule visualization tool that helps managers explore highly concurrent manufacturing schedules from various perspectives. In this technique, the big picture of the current schedule is visualized in a Gantt chart (Fig. 10(2)). Users are allowed to interactively explore the inefficiencies and reschedule manufacturing plans accordingly. PlanningVis [105] is a multi-level visualization system to support interactive exploration and comparison of production plans. This technique juxtaposes heat maps, line charts, and bar charts to visualize the differences between two plans, and thus, optimizing production plans.

6 Challenges and Opportunities

In previous sections, we summarized event sequence visualizations according to our proposed design space, extracted five analytical tasks common in visual analysis techniques for event sequences, and categorized the visual analysis techniques into three typical applications. Through this process, we combined the existing works [89], [101], [124], [140] and found the following remaining challenges in existing research and promising future research directions that are discussed in this section.

Data quality: The performance of data analysis techniques largely depends on the quality of data [52]. On top of this, the complexity of event sequence data adds difficulty to data recording and leads to more problems for data quality. Typical data quality issues include data incompleteness (e.g., missing events or timestamps), data errors (e.g., errors or inconsistency in event naming), and duplication of data records, each of which can mislead statistical analysis results. The issue of data quality implies a need for additional effort to improve data processing to prevent misleading results gathered from the source data.

Uncertainty: Uncertainty is introduced when analyzing event sequence data with quality issues or during user-specified data adjustments such as data transformation and wrangling. This uncertainty can inhibit analysts from making optimal decisions if information about uncertainty is not properly communicated in the visual analytics process [94]. Although some previous studies [21], [37] have incorporated uncertainty information in visual analytics of event sequence data, they focused on only one type of uncertainty information – the probabilistic uncertainty under an event prediction scenario. Therefore, more research is required to study the best ways of incorporating and visualizing other types of uncertainty information, such as bounded uncertainty, during the process of event sequence data analysis.

Scalability: Scalability is a well-recognized challenge in visual analytics [19], [53]. This problem becomes more significant in visual analytics of event sequence data due to the large scale (i.e., a large number of sequences) and high dimensionality (i.e., a vast number of event types) of most real-world event sequences datasets [32]. Du et al. [25] surveyed 15 strategies for sharpening analytic focus that analysts can use to reduce the data volume and pattern variety in event sequences. Some previous research touches upon this problem mainly through sequence aggregation [120] and event filtering [32], [34] to enhance the visual scalability on the sequence level and event level respectively. However, these summarization techniques hinder the inspection of detailed individual sequences and events, and the problem of how to scale across both sequence summarizations and low-level details still remains. Therefore, there is a demand for a scalable visual analytics pipeline that follows the Visual Information-Seeking Mantra by Shneiderman [100]: "overview first, zoom and filter, then details-on-demand" to allow users to flexibly switch between visual summaries and sequence details.

Heterogeneity: Event sequence data can contain a variety of heterogeneous temporal events. For example, medical health records usually include multiple event types such as diagnostic events, lab test results, vital signs, drug administrations, etc. Social media data includes multimedia content, such as text, image, and video. Events of each event type are observed or recorded with different sampling rates and show different event patterns, which leads to great difficulty for aggregating and organizing data from multiple sources. Most existing techniques choose to assemble all types of events to form a unified process for modeling and display. However, this may hinder the discovery of relationships between event types and distinctive patterns from disparate event types, which is crucial for investigative tasks and sense-making processes [117]. To solve this issue, a visual analytics framework needs to be developed, enabling both the integrated analysis of multiple event processes and the investigation of patterns for individual processes.

Multivariate event sequence visualization: Existing visual analytics techniques for event sequences generally characterize events based on their types and timestamps only. Besides these two common event attributes, however, events in a sequence can also be associated with multivariate data. For instance, lab test events in medical data are associated with specific test values, and financial transaction records also contain information about bank accounts and the monetary amount of a transaction. It still remains challenging to visualize multivariate event sequences due to the large number of event attributes a single event may include, coupled with the additional heterogeneity introduced by different data formats of the variables linked to events. Cappers and Wijk [7] provide a starting point of this issue by displaying the distributions of attributes for each individual event using lists of bar charts. However, this method can be limited for the discovery of the association between attributes of the same event or between multiple event types. This implies a need for a new visualization design that is able to represent categorical event types and multivariate attributes at the same time.

Interpretability: The chosen analysis model is a critical component in the pipeline of visual analytics [53]. In the pursuit of better analytical performance, recently developed visual analytics tools tend to leverage advanced machine learning or deep learning models with considerably high complexity. These, however, introduce issues of interpretability of the analysis results and a lack of control over the analytical process, both of which are essential for high-impact analytical tasks such as precision medicine and financial investments [17]. To address such problems, there has been an increased research investment towards explainable artificial intelligence [71], [104], with the goal of uncovering the inner workings of complex models. Even so, the mechanisms underlying these models can be difficult for non-expert users to understand. Thus, there is a high demand for visual analytics techniques that can organize, transform, and communicate model-level interpretations into comprehensible and actionable guidance. Some recent advancements [16], [38], [48] tackle this issue with a focus on a particular analytical tasks and analysis models, yet more generalizable techniques must be explored and developed.

Causality Analysis: From our review of event sequence analysis techniques, we noticed that causality analysis for event sequence data has gained increased attention in the data mining community over the past years. Many causality analysis techniques have been proposed [128], [134] to uncover the cause-and-effect relationship between events. However, very few visual analytics techniques have been developed for the causality analysis of event sequences. Despite that some existing visual analytics methods are developed for analyzing multivariate data [113], [114], the temporal nature and high dimensionality of event sequence data can lead to additional challenges, which are worth addressing in future research.

Deep Learning & Machine Learning: The capability of visual analytics techniques is largely determined by analytics techniques. As the complexity of data is exponentially increased, it is a challenging task for visual analytics techniques to process the data efficiently. To overcome this challenge, deep learning techniques, such as Recurrent Neural Networks, Variational AutoEncoders, Transformer, BERT are employed in advanced visual analysis techniques. For instance, CarePre [48] employs Recurrent Neural Networks to predict future activities based on historical event sequences. Eventthread3 [38] leverages variational autoencoders to estimate underlying normal progressions for each given sequence represented as occurrence probabilities of events along with the sequence progression. In the future study, driven by more powerful deep learning and machine learning techniques, visual analytics techniques could process more complicated data and analytics tasks.

7 Conclusion

This paper presents a survey of visual analytics approaches for event sequence data. The survey proposed a taxonomy that includes a fine categorization (design spaces) and two coarse categorizations (analysis tasks and applications) for characterizing the state-of-the-art techniques. In particular, the techniques are partitioned by five analytical tasks and five applications, and featured by their corresponding design elements in the design space. Finally, the paper discusses the remaining challenges, and points out promising future research directions. With this survey, we connect prior studies in this topic by fitting them together into our taxonomy. We hope our work could provide practitioners with an overview of the alternatives approaches, and help them find the most appropriate design components in developing an effective visual analytics solution that addresses their analytical tasks at hand.

Acknowledgments

Nan Cao is the corresponding author. This work was supported in part by NSFC 62061136003. We would like to thank our reviewers, Prof. Catherine Plaisant, Prof. Ben Shneiderman, Prof. Daniel Weiskopf, and many other readers for their valuable feedback and suggestions.

References

[1] D. Bhattacharjya, K. Shanmugam, T. Gao, N. Mattei, K. R. Varshney, and D. Subramanian. Event-driven continuous time bayesian networks. In *AAAI*, pp. 3259–3266, 2020.

[2] M. Brehmer, B. Lee, B. Bach, N. H. Riche, and T. Munzner. Timelines revisited: A design space and considerations for expressive storytelling. *IEEE TVCG*, 23(9):2151–2164, 2016.

[3] N. Cao, D. Gotz, J. Sun, and H. Qu. Dicon: Interactive visual analysis of multidimensional clusters. *IEEE TVCG*, 17(12):2581–2590, 2011.

[4] N. Cao, Y.-R. Lin, F. Du, and D. Wang. Episogram: Visual summarization of egocentric social interactions. *IEEE CG&A*, 36(5):72–81, 2015.

[5] N. Cao, Y.-R. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE TVCG*, 18(12):2649–2658, 2012.

[6] N. Cao, C. Shi, S. Lin, J. Lu, Y.-R. Lin, and C.-Y. Lin. Targetvue: Visual analysis of anomalous user behaviors in online communication systems. *IEEE TVCG*, 22(1):280–289, 2015.

[7] B. C. Cappers and J. J. van Wijk. Exploring multivariate event sequences using rules, aggregations, and selections. *IEEE TVCG*, 24(1):532–541, 2017.

[8] D. Chankhihort, B.-M. Lim, G.-J. Lee, S. Choi, S.-O. Kwon, S.-H. Lee, J.-T. Kang, A. Nasridinov, and K.-H. Yoo. A visualization scheme with a calendar heat map for abnormal pattern analysis in the manufacturing process. *Journal of Contents*, 13(2):21–28, 2017.

[9] M. Chen, A. Trefethen, R. Banares-Alcantara, M. Jirotka, B. Coecke, T. Ertl, and A. Schmidt. From data analysis and visualization to causality discovery. *Computer*, (10):84–87, 2011.

[10] Q. Chen, Y. Chen, D. Liu, C. Shi, Y. Wu, and H. Qu. Peakvizor: Visual analytics of peaks in video clickstreams from massive open online courses. *IEEE TVCG*, 22(10):2315–2330, 2015.

[11] Q. Chen, X. Yue, X. Plantaz, Y. Chen, C. Shi, T.-C. Pong, and H. Qu. Viseq: Visual analytics of learning sequence in massive open online courses. *IEEE TVCG*, 26(3):1622–1636, 2020. doi: 10.1109/TVCG.2018.2872961

[12] S. Chen, S. Chen, Z. Wang, J. Liang, Y. Wu, and X. Yuan. D-map+ interactive visual analysis and exploration of ego-centric and event-centric information diffusion patterns in social media. *ACM TIST*, 10(1):1–26, 2018.

[13] S. Chen, S. Li, S. Chen, and X. Yuan. R-map: A map metaphor for visualizing information reposting process in social media. *IEEE TVCG*, 26(1):1204–1214, 2019.

[14] Y. Chen, A. Puri, L. Yuan, and H. Qu. Stagemap: Extracting and summarizing progression stages in event sequences. In *IEEE Big Data*, pp. 975–981, 2018.

[15] Y. Chen, P. Xu, and L. Ren. Sequence synopsis: Optimize visual summary of temporal event data. *IEEE TVCG*, 24(1):45–55, 2017.

[16] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *NeurIPS*, pp. 3504–3512, 2016.

[17] J. Choo and S. Liu. Visual analytics for explainable deep learning. *IEEE CG&A*, 38(4):84–92, 2018.

[18] A. Choudhry, M. Sharma, P. Chundury, T. Kapler, D. W. Gray, N. Ramakrishnan, and N. Elmqvist. Once upon a time in visualization: Understanding the use of textual narratives for causality. *IEEE TVCG*, 2020.

[19] K. A. Cook and J. J. Thomas. Illuminating the path: The research and development agenda for visual analytics. Technical report, Pacific Northwest National Lab, 2005.

[20] T. N. Dang, P. Murray, J. Aurisano, and A. G. Forbes. Reactionflow: an interactive visualization tool for causality analysis in biological pathways. In *BMC proceedings*, vol. 9, p. S6. BioMed Central, 2015.

[21] F. Du, S. Guo, S. Malik, E. Koh, S. Kim, and Z. Liu. Interactive event sequence prediction for marketing analysts. In *ACM CHI*, pp. 1–8, 2020.

[22] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Eventaction: Visual analytics for temporal event sequence recommendation. In *IEEE VAST*, pp. 61–70, 2016.

[23] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Finding similar people to guide life choices: Challenge, design, and evaluation. In *ACM CHI*, pp. 5498–5544, 2017.

[24] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Visual interfaces for recommendation systems: Finding similar and dissimilar peers. *ACM TIST*, 10(1):1–23, 2018.

[25] 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 TVCG*, 23(6):1636–1649, 2016.

[26] N. Elmqvist and P. Tsigas. Growing squares: Animated visualization of causal relations. In *ACM Symp. on Software Visualization*, pp. 17–ff, 2003.

[27] N. Elmqvist and P. Tsigas. Animated visualization of causal relations through growing 2d geometry. *Information Visualization*, 3(3):154–172, 2004.

[28] J. A. Fails, A. Karlson, L. Shahamat, and B. Shneiderman. A visual interface for multivariate temporal data: Finding patterns of events across multiple histories. In *IEEE VAST*, pp. 167–174, 2006.

[29] X. Fan, Y. Peng, Y. Zhao, Y. Li, D. Meng, Z. Zhong, F. Zhou, and M. Lu. A personal visual analytics on smartphone usage data. *JVLC*, 41:111–120, 2017.

[30] F. Fischer, J. Fuchs, P.-A. Vervier, F. Mansmann, and O. Thonnard. Vistracer: A visual analytics tool to investigate routing anomalies in traceroutes. In *VizSec*, pp. 80–87, 2012.

[31] 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*, 28(3):238–252, 2016.

[32] D. Gotz and H. Stavropoulos. Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE TVCG*, 20(12):1783–1792, 2014.

[33] D. Gotz, S. Sun, and N. Cao. Adaptive contextualization: Combating bias during high-dimensional visualization and data selection. In *ACM IUI*, pp. 85–95, 2016.

[34] D. Gotz, J. Zhang, W. Wang, and J. Shrestha. Visual analysis of high-dimensional event sequence data via dynamic hierarchical aggregation. *IEEE TVCG*, 26(1), 2020.

[35] M. C. Goulden, E. Gronda, Y. Yang, Z. Zhang, J. Tao, C. Wang, X. Duan, G. A. Ambrose, K. Abbott, and P. Miller. Ccvis: Visual analytics of student online learning behaviors using course clickstream data. *Electronic Imaging*, 2019(1):681–1, 2019.

[36] R. Guo, T. Fujiwara, Y. Li, K. M. Lima, S. Sen, N. K. Tran, and K.-L. Ma. Comparative visual analytics for assessing medical records with sequence embedding. *Visual Informatics*, 2020.

[37] S. Guo, F. Du, S. Malik, E. Koh, S. Kim, Z. Liu, D. Kim, H. Zha, and N. Cao. Visualizing uncertainty and alternatives in event sequence predictions. In *ACM CHI*, pp. 1–12, 2019.

[38] S. Guo, Z. Jin, Q. Chen, D. Gotz, H. Zha, and N. Cao. Visual anomaly detection in event sequence data. In *IEEE Big Data*, pp. 1125–1130, 2019.

[39] S. Guo, Z. Jin, D. Gotz, F. Du, H. Zha, and N. Cao. Visual progression analysis of event sequence data. *IEEE TVCG*, pp. 1–1, 2018.

[40] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, and N. Cao. Eventthread: Visual summarization and stage analysis of event sequence data. *IEEE TVCG*, 24(1):56–65, 2017.

[41] Y. Han, A. Rozga, N. Dimitrova, G. D. Abowd, and J. Stasko. Visual analysis of proximal temporal relationships of social and communicative behaviors. In *Computer Graphics Forum*, vol. 34, pp. 51–60. Wiley Online Library, 2015.

[42] H. He, B. Dong, Q. Zheng, and G. Li. Vuc: Visualizing daily video utilization to promote student engagement in online distance education. In *CompEd*, pp. 99–105, 2019.

[43] D. Herr, F. Beck, and T. Ertl. Visual analytics for decomposing temporal event series of production lines. In *IEEE iV*, pp. 251–259, 2018.

[44] M. A. Hibbs, N. C. Dirksen, K. Li, and O. G. Troyanskaya. Visualization methods for statistical analysis of microarray clusters. *BMC Bioinformatics*, 6(1):115, 2005.

[45] O. Huisman and P. Forer. The complexities of everyday life: balancing practical and realistic approaches to modeling probable presence in space-time. In *SIRC*, pp. 155–167, 2005.

[46] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, vol. 112. Springer, 2013.

[47] W. Jentner and D. A. Keim. Visualization and visual analytic techniques for patterns. In *High-Utility Pattern Mining*, pp. 303–337. Springer, 2019.

[48] Z. Jin, S. Cui, S. Guo, D. Gotz, J. Sun, and N. Cao. Carepre: An intelligent clinical decision assistance system. *ACM HEALTH*, 1(1):1–20, 2020.

[49] Z. Jin, S. Guo, N. Chen, D. Weiskopf, D. Gotz, and N. Cao. Visual causality analysis of event sequence data. *IEEE TVCG*, 27(2):1343–1352, 2021. doi: 10.1109/TVCG.2020.3030465

[50] J. Jo, J. Huh, J. Park, B. Kim, and J. Seo. Livegantt: Interactively visualizing a large manufacturing schedule. *IEEE TVCG*, 20(12):2329–2338, 2014.

[51] N. R. Kadaba, P. P. Irani, and J. Leboe. Visualizing causal semantics using animations. *IEEE TVCG*, 13(6):1254–1261, 2007.

[52] S. Kandel, J. Heer, C. Plaisant, J. Kennedy, F. Van Ham, N. H. Riche, C. Weaver, B. Lee, D. Brodbeck, and P. Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, 2011.

[53] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. Visual analytics: Definition, process, and challenges. In *Information Visualization*, pp. 154–175. 2008.

[54] J. Kiernan and E. Terzi. Constructing comprehensive summaries of large event sequences. *ACM TKDD*, 3(4):21, 2009.

[55] B. Koldehofe, M. Papatriantafilou, and P. Tsigas. Distributed algorithms visualisation for educational purposes. In *ITiCSE*, pp. 103–106, 1999.

[56] J. Krause, A. Perer, and K. Ng. Interacting with predictions: Visual inspection of black-box machine learning models. In *ACM CHI*, pp. 5686–5697, 2016.

[57] J. Krause, A. Perer, and H. Stavropoulos. Supporting iterative cohort construction with visual temporal queries. *IEEE TVCG*, 22(1):91–100, 2015.

[58] M.-P. Kwan. Gender and individual access to urban opportunities: a study using space–time measures. *The Professional Geographer*, 51(2):210–227, 1999.

[59] B. C. Kwon, V. Anand, K. A. Severson, S. Ghosh, Z. Sun, B. I. Frohnert, M. Lundgren, and K. Ng. Dpvis: Visual analytics with hidden markov models for disease progression pathways. *IEEE TVCG*, 2020.

[60] B. C. Kwon, M.-J. Choi, J. T. Kim, E. Choi, Y. B. Kim, S. Kwon, J. Sun, and J. Choo. Retainvis: Visual analytics with interpretable and interactive recurrent neural networks on electronic medical records. *IEEE TVCG*, 2018.

[61] B. C. Kwon, J. Verma, and A. Perer. Peekquence: Visual analytics for event sequence data. In *ACM SIGKDD*, vol. 1, 2016.

[62] P.-M. Law, Z. Liu, S. Malik, and R. C. Basole. Maqui: Interweaving queries and pattern mining for recursive event sequence exploration. *IEEE TVCG*, 25(1):396–406, 2018.

[63] W. Li, M. Funk, Q. Li, and A. Brombacher. Visualizing event sequence game data to understand player's skill growth through behavior complexity. *Journal of Visualization*, 22(4):833–850, 2019.

[64] Z. Liu, H. Dev, M. Dontcheva, and M. Hoffman. Mining, pruning and visualizing frequent patterns for temporal event sequence

analysis. In *Proc. of the IEEE VIS 2016 Workshop on Temporal & Sequential Event Analysis*, pp. 2–4, 2016.

[65] Z. Liu, B. Kerr, M. Dontcheva, J. Grover, M. Hoffman, and A. Wilson. Coreflow: Extracting and visualizing branching patterns from event sequences. In *Computer Graphics Forum*, vol. 36, pp. 527–538. Wiley Online Library, 2017.

[66] 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 TVCG*, 23(1):321–330, 2016.

[67] S. Malik, F. Du, M. Monroe, E. Onukwugha, C. Plaisant, and B. Shneiderman. Cohort comparison of event sequences with balanced integration of visual analytics and statistics. In *ACM IUI*, pp. 38–49, 2015.

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

[69] H. Mei and J. M. Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *NeurIPS*, pp. 6754–6764, 2017.

[70] A. Michotte, G. Thines, A. Costall, and G. Butterworth. La causalité perceptive. *Journal de Psychologie Normale Et Pathologique*, 60:9–36, 1963.

[71] Y. Ming, S. Cao, R. Zhang, Z. Li, Y. Chen, Y. Song, and H. Qu. Understanding hidden memories of recurrent neural networks. In *IEEE VAST*, pp. 13–24, 2017.

[72] M. Monroe. *Interactive event sequence query and transformation*. PhD thesis, 2014.

[73] X. Mu, K. Xu, Q. Chen, F. Du, Y. Wang, and H. Qu. Moocad: Visual analysis of anomalous learning activities in massive open online courses. In *Eurovis*, pp. 91–95, 2019.

[74] C. Muelder, B. Zhu, W. Chen, H. Zhang, and K.-L. Ma. Visual analysis of cloud computing performance using behavioral lines. *IEEE TVCG*, 22(6):1694–1704, 2016.

[75] K. Ng, A. Ghoting, S. R. Steinhubl, W. F. Stewart, B. Malin, and J. Sun. Paramo: A parallel predictive modeling platform for healthcare analytic research using electronic health records. *Journal of biomedical informatics*, 48:160–170, 2014.

[76] P. H. Nguyen, R. Henkin, S. Chen, N. Andrienko, G. Andrienko, O. Thonnard, and C. Turkay. Vasabi: Hierarchical user profiles for interactive visual user behaviour analytics. *IEEE TVCG*, 26(1), 2020.

[77] P. H. Nguyen, C. Turkay, G. Andrienko, N. Andrienko, O. Thonnard, and J. Zouaoui. Understanding user behaviour through action sequences: from the usual to the unusual. *IEEE TVCG*, 25(9):2838–2852, 2018.

[78] C. B. Nielsen, S. D. Jackman, I. Birol, and S. J. Jones. Abyss-explorer: visualizing genome sequence assemblies. *IEEE TVCG*, 15(6):881–888, 2009.

[79] A. Perer and D. Gotz. Data-driven exploration of care plans for patients. In *ACM CHI*, pp. 439–444, 2013.

[80] A. Perer and J. Sun. Matrixflow: temporal network visual analytics to track symptom evolution during disease progression. In *AMIA Annual Symp. Proceedings*, vol. 2012, p. 716. American Medical Informatics Association, 2012.

[81] A. Perer and F. Wang. Frequence: Interactive mining and visualization of temporal frequent event sequences. In *ACM IUI*, pp. 153–162, 2014.

[82] A. Perer, F. Wang, and J. Hu. Mining and exploring care pathways from electronic medical records with visual analytics. *Journal of Biomedical Informatics*, 56:369–378, 2015.

[83] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman. Lifelines: Visualizing personal histories. In *ACM CHI*, pp. 221–227, 1996.

[84] C. Plaisant and B. Shneiderman. The diversity of data and tasks in event analytics. In *Proceedings of the IEEE VIS 2016 Workshop on Temporal & Sequential Event Analysis*, 2016.

[85] P. J. Polack, S.-T. Chen, M. Kahng, M. Sharmin, and D. H. Chau. Timestitch: Interactive multi-focus cohort discovery and comparison. In *IEEE VAST*, pp. 209–210, 2015.

[86] P. J. Polack Jr, S.-T. Chen, M. Kahng, K. D. Barbaro, R. Basole, M. Sharmin, and D. H. Chau. Chronodes: Interactive multifocus exploration of event sequences. *ACM TIIS*, 8(1):1–21, 2018.

[87] J. Qi, V. Bloemen, S. Wang, J. van Wijk, and H. van de Wetering. Stbins: Visual tracking and comparison of multiple data sequences using temporal binning. *IEEE TVCG*, 26(1):1054–1063, 2019.

[88] P. Riehmann, M. Hanfler, and B. Froehlich. Interactive sankey diagrams. In *IEEE InfoVis*, pp. 233–240, 2005.

[89] 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, 2013.

[90] A. C. Robinson, D. J. Peuquet, S. Pezanowski, F. A. Hardisty, and B. Swedberg. Design and evaluation of a geovisual analytics system for uncovering patterns in spatio-temporal event data. *CaGIS*, 44(3):216–228, 2017.

[91] J. Rogers, N. Spina, A. Neese, R. Hess, D. Brodke, and A. Lex. Composer — visual cohort analysis of patient outcomes. *Applied Clinical Informatics*, 10(02):278–285, 2019.

[92] P. Rosenthal, L. Pfeiffer, N. H. Müller, and P. Ohler. Visruption: Intuitive and efficient visualization of temporal airline disruption data. *Computer Graphics Forum*, 2013.

[93] J. Rzeszotarski and A. Kittur. Crowdscape: Interactively visualizing user behavior and output. In *ACM UIST*, p. 55–62, 2012.

[94] D. Sacha, H. Senaratne, B. C. Kwon, G. Ellis, and D. A. Keim. The role of uncertainty, awareness, and trust in visual analytics. *IEEE TVCG*, 22(1):240–249, 2015.

[95] P. Saraiya, C. North, and K. Duca. An evaluation of microarray visualization tools for biological insight. In *IEEE InfoVis*, pp. 1–8, 2004.

[96] A. Sarikaya, M. Correll, J. M. Dinis, D. H. O'Connor, and M. Gleicher. Visualizing co-occurrence of events in populations of viral genome sequences. *Computer Graphics Forum*, 2016.

[97] P. Senin, J. Lin, X. Wang, T. Oates, S. Gandhi, A. P. Boedihardjo, C. Chen, and S. Frankenstein. Time series anomaly discovery with grammar-based compression. In *EDBT*, pp. 481–492, 2015.

[98] J. Seo and B. Shneiderman. Interactively exploring hierarchical clustering results. *Computer*, 35(7):80–86, 2002.

[99] Y. Shi, Y. Liu, H. Tong, J. He, G. Yan, and N. Cao. Visual analytics of anomalous user behaviors: A survey. *IEEE Big Data*, pp. 1–1, 2020.

[100] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Symp. on Visual Languages*, pp. 336–343, 1996.

[101] B. Shneiderman and C. Plaisant. Interactive visual event analytics: Opportunities and challenges. *Computer*, 52(1):27–35, 2019.

[102] J. Slack, K. Hildebrand, T. Munzner, and K. S. John. Sequencejuxtaposer: Fluid navigation for large-scale sequence comparison in context. In *German Conference on Bioinformatics*, pp. 37–42, 2004.

[103] L. Stopar, P. Skraba, M. Grobelnik, and D. Mladenic. Streamstory: Exploring multivariate time series on multiple scales. *IEEE TVCG*, 25(4):1788–1802, 2018.

[104] H. Strobelt, S. Gehrmann, H. Pfister, and A. M. Rush. Lstmvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks. *IEEE TVCG*, 24(1):667–676, 2017.

[105] D. Sun, R. Huang, Y. Chen, Y. Wang, J. Zeng, M. Yuan, T.-C. Pong, and H. Qu. Planningvis: A visual analytics approach to production planning in smart factories. *IEEE TVCG*, 2019.

[106] G. Sun, T. Tang, T.-Q. Peng, R. Liang, and Y. Wu. Socialwave: Visual analysis of spatio-temporal diffusion of information on social media. *ACM TIST*, 9(2):1–23, 2017.

[107] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu. A survey of visual analytics techniques and applications: State-of-the-art research and future challenges. *JCST*, 28(5):852–867, 2013.

[108] J. Trümper, A. Telea, and J. Döllner. Viewfusion: Correlating structure and activity views for execution traces. In *Theory and Practice of Computer Graphics*, pp. 45–52. The Eurographics Association, 2012.

[109] F. Viégas, M. Wattenberg, J. Hebert, G. Borggaard, A. Cichowlas, J. Feinberg, J. Orwant, and C. Wren. Google+ ripples: A native visualization of information flow. In *WWW*, pp. 1389–1398, 2013.

[110] K. Vrotsou. *Everyday mining: Exploring sequences in event-based data*. Doctoral thesis, Linkoping University, The Institute of Technology, 2010.

[111] K. Vrotsou, J. Johansson, and M. Cooper. Activitree: Interactive visual exploration of sequences in event-based data using graph similarity. *IEEE TVCG*, 15(6):945–952, 2009.

[112] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao. Unsupervised clickstream clustering for user behavior analytics. In *ACM CHI*, pp. 225–236, 2016.

[113] J. Wang and K. Mueller. The visual causality analyst: An interactive interface for causal reasoning. *IEEE TVCG*, 22(1):230–239, 2015.

[114] J. Wang and K. Mueller. Visual causality analysis made practical. In *IEEE VAST*, pp. 151–161, 2017.

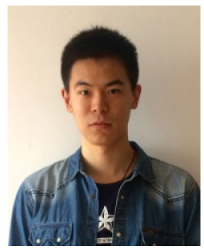
[115] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman. Aligning temporal data by sentinel events: discovering patterns in electronic health records. In *ACM CHI*, pp. 457–466, 2008.

[116] 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 TVCG*, 15(6):1049–1056, 2009.
[117] X. Wang, W. Dou, W. Ribarsky, and R. Chang. Visualization as integration of heterogeneous processes. In *Visual Analytics for Homeland Defense and Security*, vol. 7346, p. 73460B, 2009.
[118] J. Wei, Z. Shen, N. Sundaresan, and K.-L. Ma. Visual cluster exploration of web clickstream data. In *IEEE VAST*, pp. 3–12, 2012.
[119] K. Wongsuphasawat and D. Gotz. Outflow: Visualizing patient flow by symptoms and outcome. In *IEEE VisWeek Workshop on Visual Analytics in Healthcare*, pp. 25–28, 2011.
[120] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE TVCG*, 18(12):2659–2668, 2012.
[121] 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 *ACM CHI*, pp. 1747–1756, 2011.
[122] K. Wongsuphasawat and B. Shneiderman. Finding comparable temporal categorical records: A similarity measure with an interactive visualization. In *IEEE VAST*, pp. 27–34, 2009.
[123] W. Wu, Y. Zheng, K. Chen, X. Wang, and N. Cao. A visual analytics approach for equipment condition monitoring in smart factories of process industry. In *IEEE PacificVis*, pp. 140–149, 2018.
[124] Y. Wu, N. Cao, D. Gotz, Y.-P. Tan, and D. A. Keim. A survey on visual analytics of social media data. *IEEE Transactions on Multimedia*, 18(11):2135–2148, 2016.
[125] Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu. Opinionflow: Visual analysis of opinion diffusion on social media. *IEEE TVCG*, 20(12):1763–1772, 2014.
[126] C. Xie, W. Xu, and K. Mueller. A visual analytics framework for the detection of anomalous call stack trees in high performance computing applications. *IEEE TVCG*, 25(1):215–224, 2018.
[127] X. Xie, F. Du, and Y. Wu. A visual analytics approach for exploratory causal analysis: Exploration, validation, and applications. *IEEE TVCG*, 2020.
[128] H. Xu, M. Farajtabar, and H. Zha. Learning granger causality for hawkes processes. In *ICML*, pp. 1717–1726, 2016.
[129] K. Xu, Y. Wang, L. Yang, Y. Wang, B. Qiao, S. Qin, Y. Xu, H. Zhang, and H. Qu. Clouddet: Interactive visual analysis of anomalous performances in cloud computing systems. *IEEE TVCG*, 26(1):1107–1117, 2019.
[130] P. Xu, H. Mei, L. Ren, and W. Chen. Vidx: Visual diagnostics of assembly line performance in smart factories. *IEEE TVCG*, 23(1):291–300, 2016.
[131] H. Yu. Spatio-temporal gis design for exploring interactions of human activities. *CaGIS*, 33(1):3–19, 2006.
[132] X. Yuan, Z. Wang, Z. Liu, C. Guo, H. Ai, and D. Ren. Visualization of social media flows with interactively identified key players. In *IEEE VAST*, pp. 291–292, 2014.
[133] E. Zgraggen, S. M. Drucker, D. Fisher, and R. DeLine. (s—qu) eries: Visual regular expressions for querying and exploring event sequences. In *ACM CHI*, pp. 2683–2692, 2015.
[134] W. Zhang, T. K. Panum, S. Jha, P. Chalasani, and D. Page. Cause: Learning granger causality from event sequences using attribution methods. *arXiv preprint arXiv:2002.07906*, 2020.
[135] Y. Zhang, K. Chanana, and C. Dunne. Idmvis: Temporal event sequence visualization for type 1 diabetes treatment decision support. *IEEE TVCG*, 25(1):512–522, 2018.
[136] Z. Zhang, D. Gotz, and A. Perer. Iterative cohort analysis and exploration. *Information Visualization*, 14(4):289–307, 2015.
[137] Z. Zhang, K. T. McDonnell, E. Zadok, and K. Mueller. Visual correlation analysis of numerical and categorical data on the correlation map. *IEEE TVCG*, 21(2):289–303, 2014.
[138] J. Zhao, N. Cao, Z. Wen, Y. Song, Y.-R. Lin, and C. Collins. # Fluxflow: Visual analysis of anomalous information spreading on social media. *IEEE TVCG*, 20(12):1773–1782, 2014.
[139] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. Matrixwave: Visual comparison of event sequence data. In *ACM CHI*, pp. 259–268, 2015.
[140] F. Zhou, X. Lin, C. Liu, Y. Zhao, P. Xu, L. Ren, T. Xue, and L. Ren. A survey of visualization for smart manufacturing. *Journal of Visualization*, 22(2):419–435, 2019.
[141] F. Zhou, X. Lin, X. Luo, Y. Zhao, Y. Chen, N. Chen, and W. Gui. Visually enhanced situation awareness for complex manufacturing facility monitoring in smart factories. *JVLC*, 44:58–69, 2018.

Yi Guo received his M.S. degree in Financial Mathematics from the University of New South Wales, Australia in 2019. He is currently working toward his Ph.D. degree as part of the Intelligent Big Data Visualization (iDV^x) Lab, Tongji University. His research interests include data visualization and deep learning.

Shunan Guo received her Ph.D. degree in Software Engineering from East China Normal University, Shanghai, China. Her research interests include visual analytics and human-computer interaction, especially visual analytics approaches for temporal event sequences. For more information, please visit http://guoshunan.com/.



Zhuochen Jin received his B.S. degree in Computational Mathematics from Zhejiang University, China in 2017. He is currently working toward his Ph.D. degree as part of the Intelligent Big Data Visualization (iDV^x) Lab, Tongji University. His research interests include artificial intelligence and data visualization.

Smiti Kaul received her Bachelor's degrees in Computer Science and Mathematical Statistics from Wake Forest University, NC, USA. She is currently working towards an M.S. in Computer Science at the University of North Carolina at Chapel Hill, NC, USA, where she is a part of the Visual Analysis and Communication Lab.

David Gotz received his Ph.D. in Computer Science from the University of North Carolina (UNC) at Chapel Hill, NC, USA in 2005. He is currently an Associate Professor of Information Science with the School of Information and Library Science at UNC Chapel Hill. He directs the Visual Analysis and Communication Lab and conducts research on a range of topics at the intersection of data visualization, HCI, machine learning, and statistical analysis. He is also the Assistant Director for the Carolina Health Informatics Program and an Associate Member of the UNC Lineberger Comprehensive Cancer Center. He spent nearly a decade as a Research Scientist at the IBM T.J. Watson Research Center, New York, NY, USA before returning to join the UNC faculty in 2014.

Nan Cao received his Ph.D. degree in Computer Science and Engineering from the Hong Kong University of Science and Technology (HKUST), Hong Kong, China in 2012. He is currently a professor at Tongji University and the Assistant Dean of the Tongji College of Design and Innovation. He also directs the Tongji Intelligent Big Data Visualization Lab (iDV^x Lab) and conducts interdisciplinary research across multiple fields, including data visualization, human computer interaction, machine learning, and data mining. Before his Ph.D. studies at HKUST, he was a staff researcher at IBM China Research Lab, Beijing, China. He was a research staff member at the IBM T.J. Watson Research Center, New York, NY, USA before joining the Tongji faculty in 2016.