Visual Analysis of High-Dimensional Event Sequence Data via Dynamic Hierarchical Aggregation

David Gotz, Jonathan Zhang, Wenyuan Wang, Joshua Shrestha, and David Borland

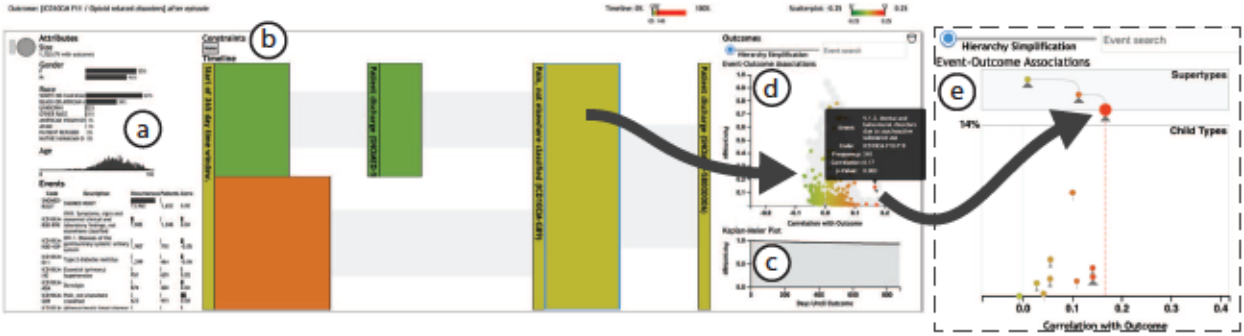

Fig. 1. The Cadence system for temporal event sequence visualization. (a) Interactive bar charts and histograms summarize non-temporal attributes and a variety of temporal event statistics. (b) An interactive flow-based timeline allows users to dynamically define and explore pathways within the temporal event data. Selections within the timeline link to (c) a Kaplan-Meier chart to summarize outcomes-over-time. Timeline selections also link to (d and e) a novel scatter-and-focus visualization which leverages dynamic hierarchical aggregation, scenting, and optimization-based layout to support navigation of high-dimensional hierarchical event data.

Abstract— Temporal event data are collected across a broad range of domains, and a variety of visual analytics techniques have been developed to empower analysts working with this form of data. These techniques generally display aggregate statistics computed over sets of event sequences that share common patterns. Such techniques are often hindered, however, by the high-dimensionality of many real-world event sequence datasets which can prevent effective aggregation. A common coping strategy for this challenge is to group event types together prior to visualization, as a pre-process, so that each group can be represented within an analysis as a single event type. However, computing these event groupings as a pre-process also places significant constraints on the analysis. This paper presents a new visual analytics approach for dynamic hierarchical dimension aggregation. The approach leverages a predefined hierarchy of dimensions to computationally quantify the informativeness, with respect to a measure of interest, of alternative levels of grouping within the hierarchy at runtime. This information is then interactively visualized, enabling users to dynamically explore the hierarchy to select the most appropriate level of grouping to use at any individual step within an analysis. Key contributions include an algorithm for interactively determining the most informative set of event groupings for a specific analysis context, and a scented scatter-plus-focus visualization design with an optimization-based layout algorithm that supports interactive hierarchical exploration of alternative event type groupings. We apply these techniques to high-dimensional event sequence data from the medical domain and report findings from domain expert interviews.

Index Terms—Temporal event sequence visualization, visual analytics, hierarchical aggregation, medical informatics

1 Introduction

Temporal event data are collected and analyzed across a broad range of domains. Reflecting this ubiquity, visual analytics techniques have been designed to support event sequence analysis across a diverse set of application areas including sporting events (e.g., [15]), career progression (e.g., [21]), transportation logistics (e.g., [20]), and large-scale system logs (e.g., [55]). These applications typically aggregate and visualize large collections of event sequences–time-ordered lists of discrete events that describe some underlying process (e.g., an athletic team's performance over a season, an individual worker's career progression, the response by emergency services to a car accident, or a user's interaction with a website). By supporting the analysis of large numbers of event sequences describing the same underlying process, these tools enable users to gain insights about common patterns, rare event paths, and associations between temporal patterns and specific performance measures.

These analytical goals, however, are often hindered by the complexity present in real-world collections of event sequence data. For example, medical data analysis has been a widely studied application of event sequence visualization techniques (e.g., [10, 15, 32, 37, 38, 43, 53, 54]). Medical analyses routinely use data from large electronic medical record systems that contain data spanning several years for millions of patients [39, 51]. Moreover, the data is represented using coding systems that have hundreds of thousands of unique types of events that can occur (diagnoses, medications, lab tests, etc.) (e.g., [1, 2]).

Early visual analytics methods focused on overcoming challenges arising from a large volume of event sequences (as opposed to large numbers of distinct events types) by aggregating identical sequences of events, and visualizing aggregate statistics via tree-based or flow-based visualizations techniques. This approach successfully enabled users to discover common patterns and their relative frequencies from a large number of sequences (e.g., [53, 54]). However, these systems

- *David Gotz is with the School of Information and Library Science at the University of North Carolina at Chapel Hill. E-mail: gotz@unc.edu.*
- *Jonathan Zhang is with the Dept. of Biostatistics at the University of North Carolina at Chapel Hill. E-mail: jzhang42@live.unc.edu.*
- *Wenyuan Wang is with the School of Information and Library Science at the University of North Carolina at Chapel Hill. E-mail: vaapad@live.unc.edu.*
- *Joshua Shrestha is with the Dept. of Computer Science at the University of North Carolina at Chapel Hill. E-mail: prayash@live.unc.edu.*
- *David Borland is with RENCI at the University of North Carolina at Chapel Hill. E-mail: borland@renci.org.*

Manuscript received 31 Mar. 2019; accepted 1 Aug. 2019.
Date of publication 16 Aug. 2019; date of current version 20 Oct. 2019.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2019.2934661

did not scale effectively to enable the analysis of high-dimensional event datasets with large numbers of distinct event types. High-dimensionality hinders many aggregation-based approaches because the large number of event types produces an even larger number of distinct event sequence patterns. Moreover, higher dimensionality typically results in increased sparsity, resulting in an increased number of smaller aggregated subgroups that limit the statistical significance associated with any patterns that are discovered.

Responding to this challenge, more recent work has led to a variety of additional coping strategies to overcome the challenge of high-dimensionality. One widely used technique, which Du et al. found in 80% (16 of 20) of the systems they surveyed [10], is *grouping events by category*. In this strategy, systems define categories of events such that all occurrences of any of the distinct event types within a category are treated as equivalent. This approach is highly effective because it directly addresses the dimensionality problem by limiting the number of event types.

However, this approach often requires very aggressive grouping (e.g., reducing hundreds of thousands of event types to dozens) and, as Du et al. observed [10], the grouping of events is typically performed as a pre-process (e.g., [31, 36]). This design choice–to decide as a pre-process which groups of event types should be treated as equivalent–is motivated in part by a very practical concern: most existing techniques have difficulty visualizing high-dimensional event sequence data. Therefore, the number of event types must be reduced before data can be loaded into the visualization system.

Unfortunately, pre-defining which sets of events are treated as equivalent is highly constraining: (1) it prevents users from interactively using the visualization tools to look at event data from multiple levels-of-detail, and (2) it requires assumptions to be made at the time of pre-processing about which categorizations are most useful to reduce dimensionality for a given analysis task. Moreover, the most appropriate level of grouping is both data- and task-dependent, suggesting that an analyst may be best served by different grouping choices at different points in the analysis process.

Motivated by the need to provide users with interactive control over event type grouping as part of the event sequence analysis process, this paper presents a new visual analytics approach for dynamic hierarchical dimension aggregation. The approach leverages a pre-defined hierarchy of dimensions to computationally quantify the informativeness, with respect to a specific measure of interest (e.g., a medical outcome), of alternative groupings within the hierarchy. This information is then interactively visualized, enabling users to dynamically explore the hierarchy to select at runtime the most appropriate level of grouping to use at any point during an analysis. While these methods are not specific to medical data, we apply the techniques to high-dimensional event sequence analysis using large-scale event type hierarchies from the medical domain.

The key research contributions presented include:

- An efficient and tunable algorithm that interactively determines, given user preferences, the most informative set of event groupings across a large-scale hierarchical set of event types for a given analytical context. The algorithm leverages a measure for quantifying the informativeness of a given event, or event grouping, with respect to a specific outcome measure of interest.
- A novel scatter-plus-focus visualization design that supports interactive hierarchical exploration of the space of event type groupings. The visualization adopts scented navigation cues to help users navigate complex hierarchies. Interactive focus control and an optimization-based layout algorithm are used to manage complexity and overcome challenges of overplotting.
- Integration of the above methods within *Cadence*, a web-based visual analytics system for population health applications that enables users to explore high-dimensional temporal event sequence datasets using dynamic hierarchical aggregation.

This paper describes the contributions enumerated above, presents an example use case, and reports findings from domain expert interviews. A discussion of these results highlights strengths of the dynamic hierarchical aggregation techniques described in this paper and motivates opportunities for future research.

2 Related Work

The contributions of this paper are most closely related to past research on event sequence visualization, event sequence analysis, hierarchical visualization, and visual scenting. In addition, the prototype application used for the case study and interviews, *Cadence*, leverages techniques for tracking provenance and selection bias.

2.1 Visualization of Temporal Event Sequences

Given the broad range of applications, as well as the unique methodological challenges posed by the volume and complexity of event sequence data, the visualization of temporal event sequences has been widely studied. Initial approaches focused on interactive alignment around sentinel events with individually drawn event sequences (e.g., [50]). These techniques were powerful, but had limited utility when working with large numbers of event sequences, as is common in many real-world applications.

Later work utilized aggregation to visualize large volumes of event sequences. This approach enabled the visualization of datasets with very large numbers of event sequences by computing aggregate data structures in tree [54] or graph [53] form, then rendering visual representations of these structures (e.g, using icicle plots [25] or Sankey-like diagrams [42], respectively) rather than individual events.

These aggregation-based approaches are, in theory, infinitely scalable for large numbers of individual sequences (in the same way that a bar chart can be used equally well to visualize a binary categorical distribution for 100 items or 1 billion items). However, the variety of data contained within many event sequence datasets–which can contain upwards of tens of thousands of unique event types–is more fundamentally challenging for visualization because the resulting increase in variation between event sequences interferes with aggregation.

This challenge has led to a variety of approaches exploring alternative coping strategies [10], including a priori event selection (choosing a small number of events to include in an analysis and ignoring the rest, e.g., [55]), dynamic event selection via user interaction (e.g., [15]), simplification by pattern substitution (e.g., [32]), and event replacement rules that account for event attributes [7]. Additional strategies related to hierarchies are reviewed in Section 2.3. These approaches typically require users to select which events to include within an analysis without any grouping of similar events, or to manually identify patterns/rules to use for substitution based on an understanding of the non-simplified data. Of these approaches, the methods in this paper are closest to dynamic event selection [15]. However, rather than simply selecting which events to include, we leverage hierarchical information to support dynamic aggregation of similar event types.

2.2 Event Sequence Analytics

The use of computational methods to help surface statistically interesting patterns or features within event sequence data has been widely explored. Moreover, the results from these efforts show that combining computational approaches with interactive user exploration can be highly effective [29]. In some cases, pattern mining algorithms have been deployed to help identify sequential event patterns with high frequency [38] or with strong associations to some attribute of the sequences (e.g., an outcome measure) [18].

Analytics have also been computed at runtime in a recursive fashion in response to user interaction. This approach has been used to support high dimensional event sequence visualization through dynamic event selection in DecisionFlow [15], to enable interwoven queries and mining that can be performed at any point within a visual analysis in MAQUI [27], and to iteratively identify groups of similar sequences [48].

The methods and prototype described in this paper adopt a similar recursive analytics approach, including dynamic event selection as in DecisionFlow and enabling multiple panels of inquiry as in MAQUI. However, the primary research contributions focus on a different challenge: recursive analytics to help users select effective event groupings (i.e., groups of event types to treat as equivalent).

Other computational approaches have focused on event sequence comparison, including projects that have compared event sequences

for clickstreams (e.g., [58]) and medical patient cohorts (e.g., [28, 29]). The prototype system described in this paper exhibits some similarities to these projects, but they are not directly relevant to the research contributions outlined in this paper.

2.3 Hierarchies

Hierarchical aggregation is widely used in visualization [11], including levels-of-detail within graph drawing (e.g., [60]) and topic evolution in text visualization (e.g., [8]). Not surprisingly, therefore, Du et. al's survey of event sequence visualization methods (which routinely employ graph- or tree-based visual representations) found that hierarchical aggregation is often used as a strategy for coping with complexity in event sequence visualization [10]. However, as Du et. al also recognized, the process typically relies on a pre-process to determine how to aggregate data, without any input from the user of the visualization and without accounting for the context of a given analysis as it unfolds.

For example, CareFlow grouped medication data into predefined classes of drugs prior to visualization [36]. Similarly, ScribeRadar mapped attributes of events to a hierarchy of event types for visualization via an icicle plot using a pre-processing step that maps raw events into pre-defined hierarchical categories [55]. In contrast, this paper describes an approach that enables interactive, user-guided aggregation at runtime using hierarchical type relationships.

While the methods of this paper are broadly applicable, the described prototype system is designed to support event sequence analysis within the medical domain. Reflecting this, we leverage existing medical coding hierarchies including ICD-10-CM [2] and SNOMED-CT [44]. These medical coding systems together include over 100,000 distinct types of medical events (e.g., diagnoses, procedures), providing a specificity that is often too fine-grained to support effective analysis without event type grouping. This is reflected by major efforts within the health informatics community to better understand and manage event groupings during analysis [5, 13, 24].

Visual navigation of hierarchies (e.g., [26]) is also closely related to the work presented in this paper. This includes techniques for navigating large and complex hierarchies or graphs using focus+context methods (e.g., [45, 46]) and techniques that focus on local neighborhoods or paths (e.g., [35, 49]). In the spirit of these techniques, the scatter-plus-focus visualization proposed in this paper uses local hierarchical structures (the path to the hierarchy root, plus all children) to enable efficient interactive navigation of the event type hierarchy.

2.4 Scenting, Provenance, and Selection Bias

A number of other related areas of research have also informed the work presented in this paper including scenting, provenance, and selection bias. Scenting is an interface technique that provides users with visual cues that help direct their interactions toward more informative subspaces of information [34]. Scented Widgets follow this approach by adorning traditional widgets (e.g., radio buttons and sliders) with visualization-based cues to facilitate informed navigation within information spaces [52]. The scatter-plus-focus visualization proposed in this paper similarly adopts a scent-based approach to help users navigate large event grouping hierarchies.

The *Cadence* prototype, meanwhile, includes features that track user-defined cohorts over time both (1) as a record of insight provenance and (2) as a means to track and communicate information about emerging selection bias during the cohort selection process. These features build upon prior work in both visual cohort selection [57] and selection bias [16, 17]. However, while these features are part of the system, they are beyond the scope of this paper and described elsewhere [6].

3 Requirements for Hierarchical Aggregation

High-dimensional temporal event sequence data can be challenging to analyze for multiple reasons. First, the large number of distinct event types produces a high amount of variance between sequences. This inhibits effective aggregation of similar sequences, a key step in the visualization process for scalable event sequence visualization techniques. Second, the sparse and fine-grained specificity of high-dimensional event data can mask patterns of interest by introducing variation between sequences due to distinct event types that should in fact be treated as equivalent for a given analysis task.

Event Type	Event Frequency	Aggregate Frequency
I50	10,739	26,153
I50.1	147	147
I50.2	-	6,071
I50.20	1,383	1,383
I50.21	635	635
I50.22	2,699	2,699
I50.23	1,354	1,354
I50.3	-	6,713
I50.30	1,896	1,896
I50.31	571	571
I50.32	2,621	2,621
I50.33	1,625	1,625
I50.4	-	2,483
I50.40	366	366
I50.41	170	170
I50.42	1,103	1,103
I50.43	844	844

Fig. 2. In a dataset of 16,983 diabetes patients, 5,084 also had a heart failure diagnosis (ICD-10 codes with the I50 prefix). A total of 26,153 heart failure codes events were observed across 14 different variants.

Fortunately, high-dimensional event data is rarely composed of entirely independent event types. Instead, metadata is often available (or can be defined) to organize events within a hierarchy that captures relationships between event types (e.g. see Section 2.3). The section presents examples from real analysis tasks in the medical domain to highlight both the challenges of high-dimensional event data and the benefits of hierarchical metadata. This section then concludes with a series of design requirements that motivate the methods in Section 4.

3.1 Medical Events

As outlined in the introduction, medical analyses are a common focus of temporal event sequence visualization tools. This is due in part to the fact that medical institutions capture large amounts of event data about patients over time as part of the normal care delivery process. Much of this data is entered into electronic health record systems (EHRs) which leverage a variety of standardized coding systems such as ICD-10-CM [2] and SNOMED-CT [1]. These coding systems provide common representations that enable the interchange of data between information systems (e.g., doctor-facing EHRs and patient portals) or parties (e.g., for medical billing), and are often used for retrospective analysis [22, 23, 40].

Medical coding systems include *hundreds of thousands* of unique types of events (e.g., diagnoses, procedures, medications). Moreover, given the large number of distinct event types, the data can be very sparse with many events occurring rarely or not at all even within large datasets. In addition, there can be significant variation in the way medical events are coded using distinct but semantically similar events.

For example, consider the statistics reported in Figure 2, which summarize the occurrence frequencies of various forms of Heart Failure (the I50 family of ICD-10 codes) within a dataset from UNC Health Care containing 16,983 diabetes patients. Nearly a third of that cohort–5,084 patients–were also diagnosed with heart failure. As shown in the figure, heart failure was represented in the dataset by one of several distinct codes (I50, I50.1, I50.20, etc.) These codes capture various forms of heart failure, e.g., diastolic, acute on chronic systolic, and several others. In all, the dataset contains 26,153 occurrences of heart failure events, with about 40% (10,739) represented as generic heart failure (ICD-10 code I.50). The remaining events are spread out across 13 other codes.

As Figure 2 shows, however, there is a hierarchical relationship between the ICD-10 codes. Depending on the circumstances, an analyst might wish to treat all versions of heart failure (all variants of the I50 code) as equivalent. At another point in the same analysis, the user may wish to distinguish between different subgroups of heart failure (I50.1 vs. I50.2 vs. I50.3 vs. I50.4) to enable comparison between subgroups. After finding something interesting in one of these groups (e.g., I50.4), an analyst may wish to compare the other I50.1/I50.2/I50.3 codes against a more detailed breakdown of I50.4 (i.e., I50.40, I50.41, I50.42, and I50.43).

This example demonstrates how single conceptual values (e.g., Heart Failure) may be spread out across multiple event types that can be

	Patients	Events	Avg. Seq. Length	Event Types
Minimum	1,732	320,711	105	11,753
Maximum	8,360	1,136,681	185	15,376
Average	4,936	701,912	151	13,997

Table 1. Summary of event sequences returned by 12 *Cadence* queries.

aggregated in many ways depending on the semantics of a particular analysis. Critically, however, this example discusses just 17 event types. As shown in Table 1, cohorts returned for realistic queries against real-world medical data can contain well over 10,000 unique event types. With an event space this large organized within a multi-level hierarchy, there is a vast space of possible event code aggregations.

3.2 Design Requirements

In prior work, the grouping of events by category as described above is typically performed as a pre-process [10]. As a result, analysts are typically forced to make educated guesses about how best to aggregate event types during the data pre-processing stage. If insights discovered during a visualization session cause an analyst to inquire about an alternative grouping, the analysis must be stopped, a new round of data pre-processing performed, and a new visualization session started on the newly processed dataset. Because of the complexity of the underlying data, each iteration of this workflow can take a significant amount of time.

Through hands-on experiences collaborating with practicing health data analysts using a variety of event sequence visualization technologies, we have observed that there is a need for more flexible, dynamic approaches that enable users to explore alternative levels of event type aggregation as an integrated part of the visual analytics workflow, rather than as a pre-process. More specifically, we have identified the following key design requirements:

- R1. Computationally determine a default context-appropriate level of hierarchical aggregation.
- R2. Provide users with high-level control over how aggressively the computational method groups events.
- R3. Provide users with the ability to explore the full type hierarchy regardless of grouping level, and to select alternative groupings.
- R4. Interactive support for R2 and R3 within users' workflows.

4 Design and Algorithms

The requirements outlined in the previous section motivate the development of new techniques for dynamic hierarchical aggregation. This section begins with a brief overview of our visual analytics system for high-dimensional event sequence analysis, focusing on aspects that support the hierarchical aggregation process. It then describes the key algorithms developed to enable these capabilities.

4.1 Visualization Design

The dynamic hierarchical aggregation techniques described in this paper have been developed within the context of *Cadence*, a visual analytics platform designed for temporal event sequence analysis.

4.1.1 Data Description and Defining Cohorts

Cadence enables users to define cohorts for analysis from large collections of longitudinal electronic health record data. In these data sources, patients are represented with a combination of non-temporal attributes (e.g., gender and race) and temporal event sequences with hundreds or thousands of events per patient, capturing several years of medical history (e.g., diagnoses and procedures from specific dates). Event type hierarchies provides multiple levels of abstraction over types of events (e.g., the ICD-10 coding hierarchy [2] for diagnosis codes).

The query interface, not shown, requires users to specify *inclusion criteria* and *outcome criteria*. The inclusion criteria specify temporal event constraints for all patients to be returned by a query (e.g., the key diagnosis or procedure events, in order, that all patients returned by a query must have in their medical record). In addition to determining which patients are returned by a query, the event constraints also define time windows of interest for each patient. The outcome criteria specify temporal event constraints used to label each patient that meets the inclusion criteria with either a good or bad outcome (e.g., a bad outcome for patients who are eventually diagnosed with a particular disease). In this way, the cohort $P = \{P_i\}$ returned in response to a user query includes a set of patients, $P_i = \{\vec{a}_i, \vec{e}_i, v_i\}$ with attributes $\vec{a}_i$, a temporal event sequence $\vec{e}_i$, and an outcome v_i. This representation is similar to outcome-labeled temporal event sequences used widely in prior work (e.g., [9, 15, 30, 53, 59]).

Symbol	Definition
$P = \{P_i\}$	A cohort of n patients
$P_i = \{\vec{a}_i, \vec{e}_i, v_i\}$	A single patient with attributes $\vec{a}_i$, temporal event sequence $\vec{e}_i$, and outcome label v_i
$\vec{v} = (v_1, v_2, \ldots v_n)$	A vector of all patient outcomes
j	An event type
$C_j = \{c_{j1}, c_{j2}, \ldots, c_{jk}\}$	The k children of event type j in the event type hierarchy
$\vec{i}_j = (i_{j1}, i_{j2}, \ldots, i_{jn})$	Binary length-n event occurrence vector for event type j
X_j^2	Informative metric for event type j

Table 2. A summary of the primary notation used throughout this paper.

4.1.2 Visual Interface

A cohort is visualized using multiple coordinated views as shown in Figure 1. First, the left sidebar in Figure 1(a) includes interactive bar charts and histograms used to summarize both categorical (e.g., gender) and continuous (e.g, age) attribute variables, respectively. Right clicking on these charts enables users to revise the cohort's inclusion criteria by applying additional attribute constraints. In addition, the left sidebar includes a sortable table of event types (e.g., both individual events such as diagnoses or procedures, and groups of these events from the type hierarchy) that occur at least once within the patient event sequences included in the cohort. Users can sort the table by the number of sequences that include the event type, the total number of occurrences of an event type (an event type can occur multiple times for one patient), and the correlation between the occurrence of an event type and the outcome label.

Two additional linked visualizations are located on the right side of the interface. This includes a Kaplan-Meier plot [41] in Figure 1(c), a traditional representation for "survival" data in the medical domain, which depicts the time of onset for the outcome variable. Finally, the right sidebar also includes a scatter-plus-focus visualization showing event type prevalence and association with outcome. The individual marks in the chart correspond to event types (both individual events, and groups of events from the type hierarchy) and are color-coded based on correlation with outcome. In normal mode (Figure 1(d)), the circles are positioned by the percentage of patients with a matching event (the y axis) and the correlation of that event type or group with the outcome (the x axis).

Critically, showing marks in the scatter plot for all possible event types and groups would suffer from severe over-plotting due to the vast number of choices, making it difficult for users to discover informative events or event groupings. To overcome this challenge, the chart first visualizes the density of all event types and hierarchical type groups (every node in the event type hierarchy) using a grayscale hexmap as a background (darker gray representing a higher number of event types). It then employs a context-dependent importance measure to define a cut through the event type hierarchy which provides the most informative level of aggregation (*R1*). Event types or groups along this cut are then visualized as marks within the scatter plot. A slider located above the plot enables users to make global adjustments as to how aggressively the importance measure is used to group events (*R2,R4*).

Clicking on any of the event type marks transitions the visualization to a focused mode (Figure 1(e)) that enables users to navigate up and down the event type hierarchy for specific event type groups (*R3,R4*). The focused view uses an optimization-based layout algorithm and scented glyphs to help users navigate the hierarchy. The focus event type is shown using an enlarged circle, with its supertypes in the type hierarchy arranged above it along an x axis that encodes correlation of the event type with outcome. Arcs connecting the event types enable users to visually trace the path to the root event type and see the changes in correlation at different levels of aggregation. The x axis is repositioned to center around the focus event type and zoomed in

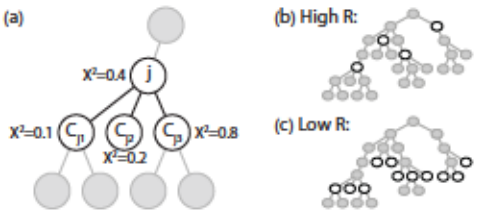

Fig. 3. (a) The most informative cut through the event type hierarchy is determined by descending until the fraction of children more informative than their parent is below a threshold R. (b) A higher R results in a higher cut. (c) A lower R will descend deeper into the hierarchy.

to a narrower extent to support more detailed comparison. Animated transitions emphasize the change in scale to users.

Below the focused event type, a focused scatter plot shows marks for all child event types (direct descendants of the focused type in the type hierarchy). The same focused x axis correlation scale is used to position the marks for both the supertypes and the child types. As a result, users are able to make easy comparisons of correlations to outcome of event type groupings moving both up and down the hierarchy. The y axis for the focused scatter plot corresponds to the proportion of event sequences containing an event type (as in the normal scatter plot). However, the top of the axis is positioned just below the focused event type mark and the maximum value for the axis adjusted to be equal to the proportion of sequences containing the focused event type. This ensures that all child event types fit within the y axis range.

Any of the supertype or child type marks can be clicked to change the focus to the corresponding event type. This enables users to precisely navigate the type hierarchy to explore alternative grouping levels (*R3,R4*). Because the focused mode only displays a small neighborhood of the full type hierarchy, visual scenting is used to help guide user exploration through the hierarchy. The scent is visualized using variable width triangles located beneath each circle. Wider triangles suggest that the subtree located below the event type in the hierarchy exhibits high heterogeneity for correlation while a narrow triangle suggests that the subtree is more homogeneous. More details about the techniques used in the scatter-plus-focus visualization are provided in Section 4.2.

At the center of the interface is an interactive event sequence timeline which follows an interactive milestone-based design similar to DecisionFlow and subsequent systems [15, 27]. In this view, *milestone events* are represented with vertical bars whose height corresponds to the proportion of patient event sequences with a given event. These vertical bars are linked with *time edges* whose width encodes the average time between milestones for sequences that contain the corresponding event types in the required order. As with milestones, the height of a time edge encodes the proportion of a cohort's patients whose sequences are represented by the edge.

Both milestones and time edges are colored by the average outcome of the corresponding patient group using a red-yellow-green color scale. This default scale uses red to represent poor outcomes and green to represent good outcomes, with the extent of the scale determined via the interactive legend above the timeline. A red-yellow-green scale is used by default to align with the preferences and familiarity of the target medical user population. However, alternative color scales are available to accommodate color-blind users.

In Figure 1(b), the timeline view displays a cohort of patients diagnosed with pain prior to being discharged from the hospital. In addition, the visualization shows data for one year prior to the pain diagnosis. The parallel paths at the start of the timeline show that slightly fewer than half of these patients were also discharged from a hospital visit in the year prior to the pain diagnosis, and that those patients had better outcomes than their "not hospitalized before pain diagnosis" peers.

Importantly, the timeline visualization enables the user to select either milestones or time edges to view more details about the patients and events that they represent. For milestones, data is displayed about events that occur at the same time as a selected milestone event. For time edges, data is displayed for all events that occur in between the pair of milestones that define the edge. Upon a new selection within the timeline, the charts in both sidebars are updated to show corresponding data. Most critically, each time the selected section of the timeline changes, the scatter-plus-focus visualization is recomputed with an updated set of event groupings for the current context. Animated transitions combined with event type search and highlighting enable users to compare event statistics across different timeline elements. Finally, users can select an appropriate event type or type grouping from the scatter-plus-focus visualization to be added as a new milestone in the timeline. Following the pattern of prior work [15], adding a new milestone enables exploratory analysis by causing the creation of new time edges for which statistics are recursively calculated and visualized.

4.2 Key Algorithms and Interactions

The interface described above leverages three key algorithmic solutions to support interactive control over the hierarchical event type grouping process. This section defines the three algorithms and describes how they are used during visualization.

4.2.1 Informativeness Measure for Hierarchical Aggregation

The scatter-plus-focus visualization provides users with information about both the prevalence of different event types, and the association between the patient outcome and occurrence of those event types. However, many event types have very low frequencies of occurrence which makes them less informative when looking for statistically meaningful patterns. An event type hierarchy can aggregate these low level events into higher level groups, resulting in fewer events with higher frequencies. However, too much aggregation results in a loss of the information that analysts are seeking in their analysis (e.g., aggregating all the way to the generic root event of a hierarchy would eliminate all distinguishing information between events). In summary, visualizing events with either too little or too much aggregation can result in loss of information. We therefore define an informativeness measure which we evaluate for every node in the event type hierarchy each time the user's analytic context changes (i.e., via selections or the creation of new milestones in the timeline). The results are used to determine a cut through the event type hierarchy representing the optimal (in terms of the measure) level of aggregation. The optimal groupings are then visualized (by default) within the scatter plot.

Measure Definition. Given a specific cohort $P = \{P_i\}$ with events $\vec{e}_i$ and outcome v_i for each patient P_i, an informative measure based on the chi-square statistic is computed for every event type in the hierarchy. We define a binary outcome vector for the cohort as follows:

$$\vec{v} = (v_1, v_2, \ldots, v_n) \tag{1}$$

where $v_i = 1$ if the patient has the outcome or $v_i = 0$ otherwise.

We then define a binary event type vector $\vec{t}_j$ for each event type j in the event type hierarchy as follows:

$$\vec{t}_j = (t_{j1}, t_{j2}, \ldots, t_{jn}) \tag{2}$$

where $t_{ji} = 1$ if either type j or a subtype of j occurs at least once within $\vec{e}_i$, or $t_{ji} = 0$ otherwise. We emphasize that values in this vector are set equal to 1 if the type t or any of its subtypes within the event type hierarchy are observed for a given patient. Therefore, when j is the root event type at the top of the hierarchy, the vector $\vec{t}_j$ will be a vector of all ones because all events are subtypes of the root event type.

We then tabulate a contingency table based on the two previous binary vectors (each with length N) as follows:

Event \ Outcome	0	1	Total
0	n_{00}	n_{01}	$n_{0.}$
1	n_{10}	n_{11}	$n_{1.}$
Total	$n_{.0}$	$n_{.1}$	N

where for event type j, $n_{ab} = \sum_{i=1}^{N} (I[t_{ji} = a] I[v_i = b])$ and I being the indicator function. We calculate a statistic based on the chi-square statistic for independence with a Yates correction for continuity:

$$X_j^2 = \frac{N(max(|n_{00}n_{11} - n_{01}n_{10}| - N/2, 0))^2}{n_{.0}n_{.1}n_{0.}n_{1.}} \tag{3}$$

where $X_j^2 \sim \chi_1^2$ [4], and $X_j^2 = 0$ if any term in the denominator is zero. This measure reflects the strength of the association between an

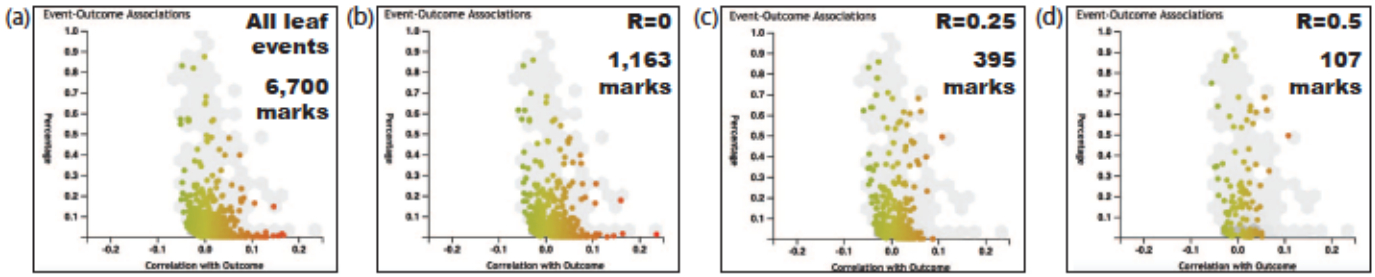

Fig. 4. The level of aggregation is controlled by R, with higher R values resulting in more aggressive event type grouping and fewer visual marks.

event type and the outcome, with larger values representing a stronger association. Yates correction is used to prevent overestimation of rare event types (common in sparse data) due to assumptions behind the chi-square statistic [56]. The chi-square statistic was selected as the basis of this measure because it is well-suited to quantify the strength of association between two binary variables sampled from the same population, and because it gives a p-value. However, other methods, such as the Jaccard Index or other set-based statistics, could be used as the basis for alternative measures and could fit naturally within our overall framework.

Selecting Informative Event Type Groupings. Given a hierarchy of event types, and the importance measure X_j^2 which can be applied to any type j within the hierarchy, we next define an algorithm that determines a cut through the type hierarchy such that the selected event types represent the most informative level of event groupings (*R1*).

The algorithm for determining the most informative cut recursively traverses the event type hierarchy starting from the root event type. At each event type visit, the algorithm compares event type j with all of j's children in the hierarchy. If j is determined to be more informative than its combined children, then type j is selected as part of the cut and its descendants in the hierarchy are no longer considered. If the child event types are considered more informative, the the process is recursively applied to each of the children one at a time to descend further into the hierarchy. If the algorithm reaches a leaf node in the hierarchy, the leaf node is considered part of the most informative cut.

The key decision point in this algorithm is the comparison between the informativeness of type j and the informativeness of its children $C_j = \{c_{j1}, c_{j2}, \ldots c_{jn}\}$. Because C_j can contain more than one event type, we define a one-to-many comparison criterion R_j which is based on the proportion of children event types C_j for which the informative measure exceeds the informativeness of j. More formally:

$$R_j = \frac{\sum_{i=1}^{|C_j|} I[X_j^2 < X_{c_{ji}}^2]}{|C_j|} \quad (4)$$

where c_{ji} is the ith child of event type j. Type j is classified as informative if either (1) it has no children or (2) $R_j \leq R$, where $0 \leq R \leq 1$ can be specified by the user ($R = 0$ being the default) via a slider in the user interface (*R2,R4*). After determining the most informative cut, we then select events in the cut where $X_j^2 > 0$. Intuitively, a lower value of R generally selects events that are deeper into the event type hierarchy because it reflects a stricter criterion for stopping the hierarchy traversal algorithm. This results in less aggregation of event types, and therefore a larger number of lower level informative events. In contrast, a higher R will result in a cut of informative events that is higher in the event type hierarchy. This produces a cut with fewer and higher level event types that result in a greater amount of aggregation. This effect is illustrated in Figure 3.

The effect of adjusting the R threshold with a realistic dataset containing an event hierarchy with 13,118 event types is shown in Figure 4. The figure's first panel shows the result from choosing all leaf nodes in the type hierarchy (event types without children), which is the approach followed in prior work. The other three panels depict the results when using R values of 0, 0.25, and 0.5 At the most aggressive level of aggregation, only 107 event types are displayed in the scatter plot. This threshold can be adjusted interactively by users via the Hierarchy Simplification slider located above the scatter plot in Figure 1(d,e).

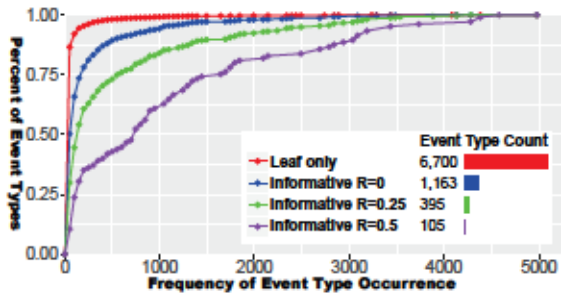

Fig. 5. Using only leaf event types results in many low-frequency event types. As this data from a test use case shows, hierarchical aggregation results in fewer and higher-frequency event types.

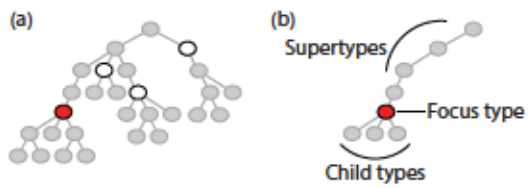

Fig. 6. (a) Clicking on an event type in the scatter-plus-focus visualization transitions the chart to a focused mode. (b) The focused mode displays the selected event type, its children, and all supertypes up to the root of the hierarchy. All other types are hidden from view.

Figure 5 provides a deeper look at the effect of aggregation at these four different configurations (all leaves, R=0, 0.25, and 0.5). These charts show that nearly all event types in Leaf Only mode have very few occurrences. This makes the detection of meaningful patterns extremely difficult. Increasing R values result in increasing amounts of aggregation that produce fewer event types with far greater frequency of occurrence.

We note that X_j^2 is influenced by the expected variance of event type occurrence frequencies in real world datasets. Somewhat analogous to false negatives in traditional statistical measures, random fluctuations in these frequencies can potentially impact the selected event type grouping level. While computational techniques could potentially limit this effect (e.g., [19]), the impact is mitigated by the fact that the cut provides only a starting point for aggregation, with users able to interactively explore alternative levels of grouping.

4.2.2 Optimization-Based Layout for Focused View

When users click on an event type mark in the scatter-plus-focus visualization, the chart transitions to a focused mode that enables users to explore up and down the type hierarchy, as described in Section 4.1.2. More specifically, the focused mode displays the focused event type, its supertypes up through the root of the type hierarchy, and all immediate children (see Figure 6).

Critically, navigating up and down the event type hierarchy requires users to see and directly interact with (via clicking) the individual event type circles in the focused chart. To overcome challenges of overplotting (see Figure 7), a dual-view design was adopted.

This design positions the focused event type and its ancestors above an optimization-based scatter plot of the focused type's children. As

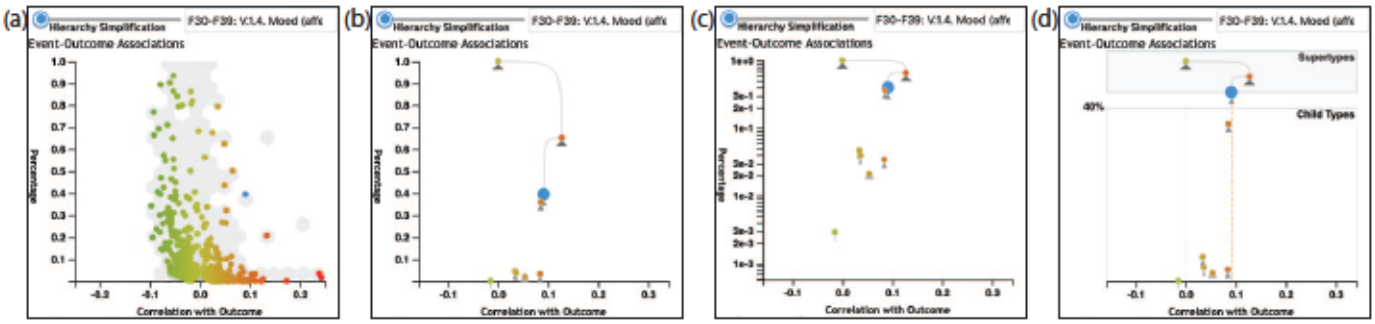

Fig. 7. Clicking on an event type in (a) the scatter-plus-focus visualization transitions to a focused view. (b) Maintaining the axes in the focused view results in overplotting, especially for the large number of low frequency events common to high-dimensional event sequences. (c) A log scale reduces overplotting for low frequency event types, but worsens problems for more frequent events and makes interpretation less intuitive. (d) An optimization-based layout maintains an intuitive scale while overcoming challenges of overplotting.

shown in Figure 7(d), the top of the chart starts the path from the root event type, through all supertypes, to the focused event type. The y axis in this portion of the chart maps to the depth of the event type in the type hierarchy. Below the focused type is a scatter plot of all children using an x axis that is centered on the correlation value for the focused type and aligned between both portions of the dual-view chart. Vertical guide lines depict both zero correlation and the focused event type's correlation value to facilitate comparison between types.

The y axis positions for the marks within the child type scatter plot are determined by an optimization-based layout algorithm that aims to balance two competing layout priorities: (1) marks should be positioned as close as possible to their ideal scatter plot location within the y axis scale, and (2) marks should not overlap. For this reason, y axis positions closely approximate the actual percentage of event sequences that contain the corresponding event type, with adjustments made to avoid overplotting.

For a focus event type j, the layout process begins by assigning every child event type C_{ji} to an initial position (x_i, y_i) using the scatter plot's axis scales. Then, vectors $\vec{x}$ and $\vec{y}$ are defined as the initial x and y positions for all marks, respectively. The values in $\vec{x}$ are held constant and used to render the marks within the visualization. However, the y positions are adjusted via a minimization process to obtain an optimized set of y positions $\vec{y'}$. The optimization minimizes the following cost function:

$$\begin{aligned} \underset{\vec{y'}}{\operatorname{argmin}} f(\vec{y_i}) &= \alpha \sum_{i=1}^{|\vec{y'}|} \sum_{j=1}^{|\vec{y'}|} \Omega(y'_i, x_i, y'_j, x_j) + (1-\alpha) \sum_{i=1}^{|\vec{y'}|} |y'_i - y_i| \\ \textit{where} \quad & \Omega(x_1, y_1, x_2, y_2) = \max\left(0, d - \sqrt{(x_1-x_2)^2 + (y_1-y_2)^2}\right) \\ \textit{s.t.} \quad & y_{min} \le y'_i \le y_{max} \quad \forall y'_i \in \vec{y'} \\ & \operatorname{sgn}(y'_i - y'_j) = \operatorname{sgn}(y_i - y_j) \quad \forall i, j \in [1, |\vec{y'}|] \end{aligned} \tag{5}$$

where d is the diameter of a mark in the plot, and α is a tuning parameter used to balance between the competing elements of the cost function (overlap, and y-position distortion).

Intuitively, the cost function includes two terms. First, an overlap term sums the amount of spatial overlap between all pairs of marks. A zero-cost layout would have no overlapping marks. Second, a distortion term sums the amount of y-scale displacement applied to each mark. A zero cost layout would have no distortion to the vertical position of the marks. The relative importance of these terms is controlled by a weighting factor, α, which we set to 0.8 based on empirical observations that show it performs well in most cases.

Two constraints are applied during the optimization process. First, an extent constraint requires that the optimized y axis positions ($\vec{y'}$) fall within the extent of the y-axis scale(y_{min}, y_{max}). This ensures that the optimized positions remain within the y-axis bounds of the chart. Second, an ordering constraint ensures that the original ordering of marks along y axis is preserved in the final resulting layout (i.e., marks that occur more frequently are located above marks that occur less frequently).

Fig. 8. Scented glyphs help users efficiently navigate the type hierarchy.

The effect of the optimization can be seen when comparing the lower regions of Figure 7(b,d). The marks in the optimized version in (panel d) are more clearly separated out along the bottom of the chart. This result increases legibility with less overplotting even when compared to the log scale version (panel c), while maintaining the approximate location of these marks along the bottom of the chart to correctly communicate that the corresponding event types rarely occur.

4.2.3 Scenting

The focused mode visualization includes marks for the focused event type, its supertypes, and its direct children. Users can click on any super- or child type to change focus and navigate up or down the type hierarchy. To help guide the user towards more interesting event types within the hierarchy during interaction, a scent value is calculated for all events and rendered as part of the glyph used to represent each event type. Intuitively, the scent is designed to highlight event types whose descendants in the type hierarchy have a wide range of correlations to outcome. This property of an event type suggests that users might be interested in a lower level of aggregation that better separates event sequences by outcome. Conversely, event types whose descendent types have more homogeneous associations with the outcome would be less valuable to disaggregate The glyph for communicating the scent value is illustrated in Figure 8. Event types that are at the bottom of the type hierarchy with no children cannot be expanded, and therefore are displayed with the "no scent" indicator instead of the normal triangular representation.

The scent value for a given event type j is computed from the correlation values for the entire subtree of event types below j in the type hierarchy (not just the immediate children). The scent value is computed recursively, beginning with type j's immediate children. The difference between the maximum and minimum correlation values for the event type's children is determined. Then, the maximum scent for each of the children individually is determined. The scent for j is then equal to the maximum of either (1) the difference in correlation for j's children, or (2) the maximum scent for j's children. Leaf event types (with no children) are given a scent of zero by definition. As a result, the scent is equal to the maximum difference in correlation values for any set of peer event types within the subtree under j.

This value is displayed as a glyph below each event in the outcomes view when focused on a specific event. As illustrated in Figure 7, the size of the glyph describes the magnitude of this scent value. There is a separate unique visual glyph if that event is an event without children. Although this scent value describes the heterogeneity in correlation values to provide the user with a general sense of what events have the most diverse children, it does not specify what level of aggregation

the most diverse correlations are observed on. The user can click on an event to see which children of that event carry the diversity of correlation and make further inferences from there.

5 Example Use Case and Domain Expert Interviews

This section presents a use case that demonstrates how dynamic hierarchical aggregation can help support high-dimensional event sequence analysis tasks, and provides insights about the utility of these methods for health applications through domain expert interviews.

5.1 Example Use Case

Recognizing the value of analyzing and learning from medical practice, medical institutions and governments have invested in building large collections of of electronic medical data (e.g., [3, 12, 33]) to support retrospective analyses that can inform policy making, quality of care, and medical understanding. These "real-world evidence" efforts require analysts to sift through large and high-dimensional medical data to learn about which patterns or processes are associated with differences in medical endpoints, costs, or other outcomes of interest. This has led to a growing interest in visual tools that can support interactive discovery over such data [14].

Consider the use case of an analyst working with data from our institution's own clinical data warehouse [47] in an attempt to understand factors that are linked to opiate related disorders (i.e., a group of diagnoses related to addition or abuse). Today, analysts working with this data would need to make a data requests from the data warehouse managers, who in response would return data files (e.g., SAS or CSV format) with extracts from the larger database. A user would then need to manually pre-process these data extracts by cleaning and aggregating the tens-of-thousands of unique event types into a form that could be analyzed with prior visualization tools. A hunch that alternative groupings would be more effective would require the analyst to leave the visualization environment, re-process the data files to generate the alternative groupings, and then re-visualize. This process can be slow and cumbersome, interrupting the problem solving process with complex data manipulation tasks.

In contrast, Figure 9 shows screenshots of how the system can be used for this type of analysis task. The analysis begins with a query showing 1,732 patients who were discharged from a hospital after previously (during or before the hospitalization) having been diagnosed with pain. The timeline in panel (a) shows that the query also returned one years worth of events prior to the pain diagnosis. Seven percent of the patients develop opiate related disorders after being discharged from the hospital.

Clicking the timeline segment representing time between the pain diagnosis and hospital discharge, the scatter-plus-focus chart is updated with the relevant statistics. Nicotine dependence (a group of diagnosis codes with several subtypes) is a clear outlier. It is quite prevalent (360 of 1,732 patients) and has a relatively strong correlation with opiate disorders ($\rho = 0.13$). Clicking on the circle for this event type transitions the visualization to the focused mode shown in panel (b). Here, users can see that there is a single child with very low scent. However, moving up the hierarchy, the parent type has even stronger correlation and a large scent. A tooltip reveals that this is a broader category of substance abuse (more than just nicotine), so the user clicks on the circle to revise the focus (panel c).

The animated transition shows clearly that nicotine is this event type's child with the highest frequency (it is highest along the y axis), but not the most correlated with the outcome. There are two less frequent subtypes of substance abuse that are relatively rare but exhibit higher correlation. Due to the low frequency of those types, however, the user decides to stick with the higher level substance abuse event type grouping. The user locks the selection to this event type (highlighting it in blue) and clicks the chart background to return to the non-focused mode (panel d).

Selecting the timeline segment prior to pain diagnosis, the scatter-plus-focus chart updates with new statistics and an updated set of most-informative event type groups (panel e). The user finds that the grouping remains associated with poor outcomes, but is less prominent (just 240 patients) and has weaker correlation. The user therefore returns to the post-pain diagnosis timeline segment and adds the substance abuse event type group as a new milestone. This results in an updated timeline (panel f) which enables the user to continue searching. In this case, the user finds heart procedures as a frequent and high-correlation event type group. Clicking on the corresponding circle, the focused chart shows that a more specific subtype has nearly the same frequency with high scent (panel g). The user can click down through several layers of the type hierarchy to discover that these are primarily (138 of the 148 heart procedures) ECG procedures (panel h). This event could also be added to the timeline, or the user could continue exploring the data.

5.2 Domain Expert Interviews

To gather qualitative feedback regarding dynamic hierarchical aggregation, we conducted hands-on demonstrations of the system and conducted semi-structured interviews with three medical experts. The three participants were health-focused researchers with data analysis experience. One participant was a medical doctor with joint clinical-research responsibilities, while the other two participants were PhD-level researchers. Moreover, all three participants had prior experience with i2b2 [33], an NIH-funded web-based cohort selection environment that has been deployed to support data-driven health research at our university and at many other institutions around the world. Experience with i2b2 was a requirement for recruitment to ensure that participants were within the target user population for the *Cadence* system. Each participant was interviewed independently, meeting with two study moderators for a one hour session. During the hour, the participants were given a brief introduction, followed by a demonstration of the system's key features. The participants were then interviewed by the study moderators in a semi-structured interview format. During the interview, ad hoc data exploration was encouraged in response to participant curiosity.

5.2.1 Thematic Analysis of Interview Findings

The domain experts provided wide-ranging feedback during the interview sessions, addressing a variety of features. We present a thematic analysis of the interview findings, focusing on the themes that are most directly relevant to the methods presented in this paper.

Training required. The users agreed unanimously that training is required to use the system. One mentioned that it seemed complicated when they first saw the interface, but after being oriented "it made sense." About the need for training, one expert emphasized "I don't see a problem with that." Training is required for the existing i2b2 system, for example: "i2b2 was complicated at first" as well. Speaking about both i2b2 and *Cadence*, an expert remarked that these "are for skilled users." Said another, the person using this is a "superuser...willing to put in the time to learn the interface." One expert summarized this theme as follows: this is "very cool, but there is a learning curve."

The challenge of discoverability was mentioned by one user. "People just have to know how to use [it].... They have to know they can manipulate the scroll bar to adjust the hierarchy." Similarly, the user asked "how do I know if I can click" on a circle in the scatter plot? Training can help, but this is also an opportunity for future interface enhancements.

Benefits of automated selection of aggregation level. The automated approach to suggesting the most informative level of aggregation "was very useful." Another expert remarked that "yes, [it was] very helpful," to have the system suggest a starting point. One expert mentioned that a danger in an automated approach is that there was the potential that it would restrict the ability to explore the data. However, they didn't see that as a problem because "you can control what level of aggregation is used" referencing the slider which is mapped to the R threshold described in Section 4.2.1. They mentioned that the slider enabled them to "control the simplification," and that it was very valuable and intuitive.

Intuitive navigation of the type hierarchy. Users described the focused mode as "intuitive" and "easy to interpret" after brief training. They found the hierarchy concept for event types natural, and intuitively understood what moving up or down that hierarchy meant in terms of

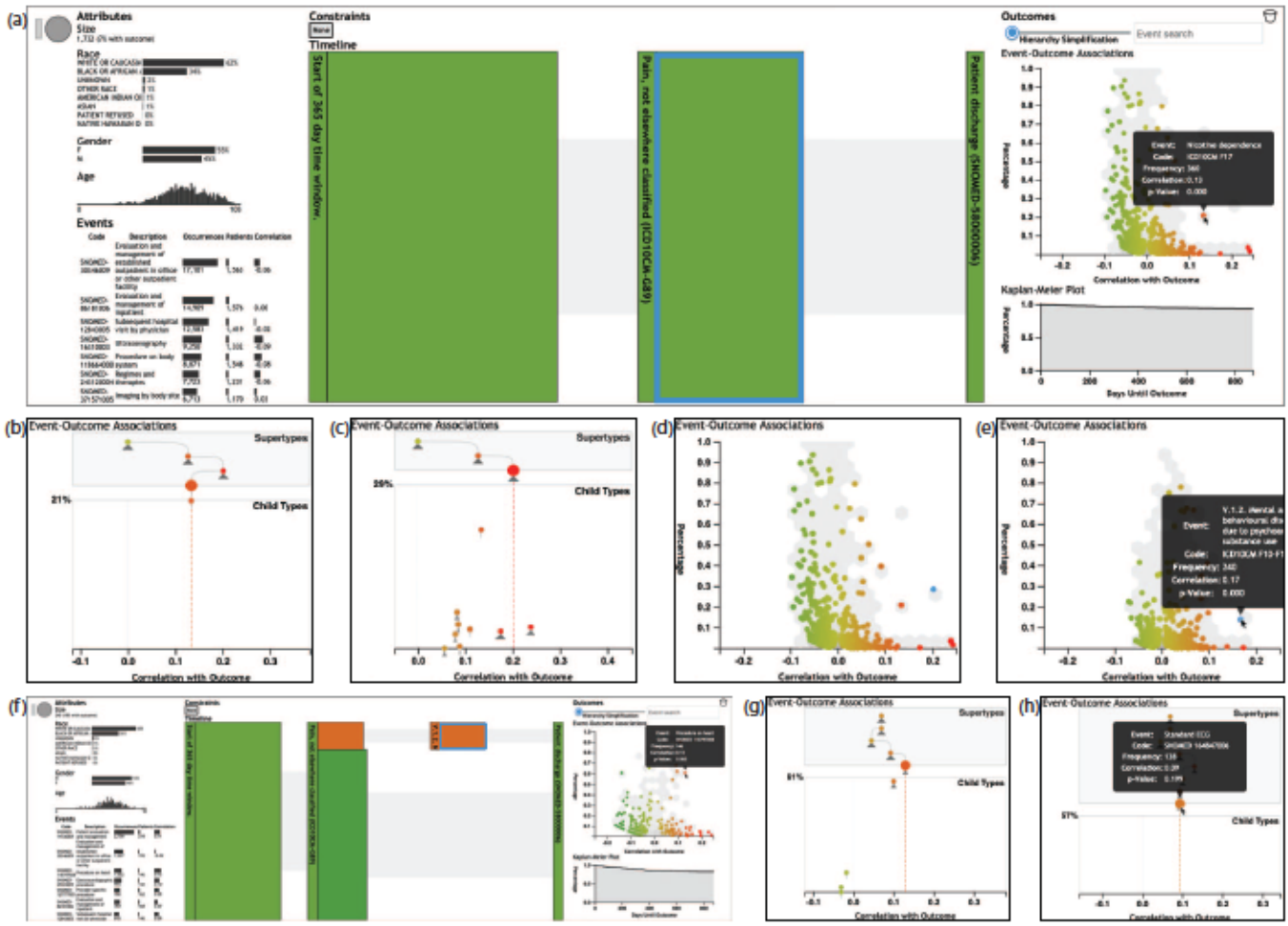

Fig. 9. Screenshots from the use case described in Section 5.1.

the analysis. "The ability to go up in the hierarchy is really useful," mentioned one user. After some exploration of the visualization on the screen at the time, another expert remarked "clearly, different types of mental illness have different correlations with discharge. I found that very useful."

With respect to the scenting, one expert found it "very helpful." Another remarked that it was "intuitive and easy to interpret" after the brief training provided during the interview session. "I liked that... It was good." The experts found it easy to use the scenting to find where interesting variance was located within the type hierarchy. There was a request, however, to display more detailed information about what led to a high scent score in response to a mouseover to avoid having to actually visit each highly scented event type.

Overall value of the approach. A number of non-feature-specific comments were made more generally about the approach. These were typically very positive as the prototype system provided many clear benefits to the domain experts. "I really like your design." Said another user: "This is very useful... for cohort studies." It enables you to "instantly generate insights" and is a powerful "hypothesis generating application." One expert highlighted a discovery during the interview session: when we "saw the spike in the age distribution, [I asked] why?" The system lets you look into it right away.

This is a "really powerful analytical tool" said one expert. "This could be a powerful tool" said another, "it has all the function that people could imagine.... I didn't know that a user interface could do this much. Could go this deep. [Could help you] choose event levels."

6 Conclusion

This paper presented a new visual analytics approach for dynamic hierarchical dimension aggregation during high-dimensional temporal event sequence analysis. It overcomes limitations in prior work by enabling users to interactively and intuitively explore and define group type aggregations as part of the analysis workflow rather than as a pre-process. The approach leverages a pre-defined event type hierarchy to computationally quantify the informativeness of alternative levels of grouping given the current analytical context. This information is then visualized to give users the ability to interactively explore alternative groupings and select the most appropriate level of grouping to use at any individual step within an analysis. This is made possible via (1) a measure of informativeness that can be applied to individual event type groupings; (2) an efficient and tunable algorithm for determining the most informative levels of aggregation from within a large type hierarchy; and (3) a novel scatter-plus-focus visualization with scenting and optimization-based layout to help users explore the type hierarchy to compare alternative levels of aggregation. Although these contributions have been implemented and evaluated (through domain expert interviews) as part of a medical data analysis tool, there is potential applicability to a broader range of similar problems.

While the results are promising, there remain several areas for future work. For example, visualization methods that afford more flexible groupings beyond those strictly defined within the type hierarchy would very useful within the medical context. Leveraging ontologies rather than tree-based hierarchies could help support this type of flexibility. Another possible topic for future work is making it easier to re-use groupings from one part of an analysis in later stages. Consistent grouping is often important, and better interactive support for this would be valuable. Interface improvements to support discoverability of advanced features would also be useful.

Acknowledgments

The research reported in this article was supported in part by a grant from the National Science Foundation (#1704018).

REFERENCES

[1] Snomed - 5-step briefing. https://www.snomed.org/snomed-ct/five-step-briefing. Accessed: 2019-03-04.

[2] ICD - ICD-10-CM - International Classification of Diseases, Tenth Revision, Clinical Modification, July 2018.

[3] A. P. Abernethy, L. M. Etheredge, P. A. Ganz, P. Wallace, R. R. German, C. Neti, P. B. Bach, and S. B. Murphy. Rapid-Learning System for Cancer Care. *Journal of Clinical Oncology*, 28(27):4268–4274, Sept. 2010. doi: 10.1200/JCO.2010.28.5478

[4] A. Agresti. *Categorical Data Analysis*. John Wiley & Sons Inc., New Jersey, 2nd ed., 2002.

[5] O. Bodenreider, D. Nguyen, P. Chiang, P. Chuang, M. Madden, R. Winnenburg, R. McClure, S. Emrick, and I. DSouza. The NLM Value Set Authority Center. *Studies in health technology and informatics*, 192:1224, 2013.

[6] D. Borland, W. Wang, J. Zhang, J. Shrestha, and D. Gotz. Selection Bias Tracking and Detailed Subset Comparison for High-Dimensional Data. *IEEE Transactions on Visualization and Computer Graphics*, 26(1), 2020.

[7] B. C. M. Cappers and J. J. v. Wijk. Exploring Multivariate Event Sequences Using Rules, Aggregations, and Selections. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):532–541, Jan. 2018. doi: 10.1109/TVCG.2017.2745278

[8] W. Cui, S. Liu, Z. Wu, and H. Wei. How Hierarchical Topics Evolve in Large Text Corpora. *IEEE transactions on visualization and computer graphics*, 20(12):2281–2290, Dec. 2014. doi: 10.1109/TVCG.2014.2346433

[9] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. EventAction: Visual analytics for temporal event sequence recommendation. In *2016 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 61–70, Oct. 2016. doi: 10.1109/VAST.2016.7883512

[10] F. Du, B. Shneiderman, C. Plaisant, S. Malik, and A. Perer. Coping with Volume and Variety in Temporal Event Sequences: Strategies for Sharpening Analytic Focus. *IEEE Transactions on Visualization and Computer Graphics*, 23(6):1636–1649, June 2017. doi: 10.1109/TVCG.2016.2539960

[11] N. Elmqvist and J. Fekete. Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2010. doi: 10.1109/TVCG.2009.84

[12] R. L. Fleurence, L. H. Curtis, R. M. Califf, R. Platt, J. V. Selby, and J. S. Brown. Launching PCORnet, a national patient-centered clinical research network. *Journal of the American Medical Informatics Association: JAMIA*, 21(4):578–582, Aug. 2014. doi: 10.1136/amiajnl-2014-002747

[13] S. Gold, A. Batch, R. McClure, G. Jiang, H. Kharrazi, R. Saripalle, V. Huser, C. Weng, N. Roderer, A. Szarfman, N. Elmqvist, and D. Gotz. Clinical Concept Value Sets and Interoperability in Health Data Analytics. In *AMIA Annual Symposium Proceedings*, 2018.

[14] D. Gotz and D. Borland. Data-Driven Healthcare: Challenges and Opportunities for Interactive Visualization. *IEEE Computer Graphics and Applications*, 36(3):90–96, 2016.

[15] D. Gotz and H. Stavropoulos. DecisionFlow: Visual Analytics for High-Dimensional Temporal Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1783–1792, 2014. doi: 10.1109/TVCG.2014.2346682

[16] D. Gotz, S. Sun, and N. Cao. Adaptive Contextualization: Combating Bias During High-Dimensional Visualization and Data Selection. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, IUI '16, pp. 85–95. ACM, New York, NY, USA, 2016. doi: 10.1145/2856767.2856779

[17] D. Gotz, S. Sun, N. Cao, R. Kundu, and A.-M. Meyer. Adaptive Contextualization Methods for Combating Selection Bias During High-Dimensional Visualization. *ACM Trans. Interact. Intell. Syst.*, 7(4):17:1–17:23, Nov. 2017. doi: 10.1145/3009973

[18] D. Gotz, F. Wang, and A. Perer. A methodology for interactive mining and visual analysis of clinical event patterns using electronic health record data. *Journal of biomedical informatics*, Jan. 2014. doi: 10.1016/j.jbi.2014.01.007

[19] D. Gotz, W. Wang, A. T. Chen, and D. Borland. Visual Model Validation Via Inline Replication. *Information Visualization*, Online First, Jan. 2019.

[20] J. A. Guerra-Gomez, K. Wongsuphasawat, T. D. Wang, M. Pack, and C. Plaisant. Analyzing Incident Management Event Sequences with Interactive Visualization. In *Proceedings of the Transportation Research Board 90th annual meeting*. The National Academies, Washington, DC, 2011.

[21] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, and N. Cao. EventThread: Visual Summarization and Stage Analysis of Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):56–65, Jan. 2018. doi: 10.1109/TVCG.2017.2745320

[22] W. Hersh, J. Cimino, P. Payne, P. Embi, J. Logan, M. Weiner, E. Bernstam, H. Lehmann, G. Hripcsak, T. Hartzog, and J. Saltz. Recommendations for the Use of Operational Electronic Health Record Data in Comparative Effectiveness Research. *eGEMs (Generating Evidence & Methods to improve patient outcomes)*, 1(1), Oct. 2013. doi: 10.13063/2327-9214.1018

[23] P. B. Jensen, L. J. Jensen, and S. Brunak. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405, June 2012. doi: 10.1038/nrg3208

[24] G. Jiang, R. Kiefer, E. Prudhommeaux, and H. R. Solbrig. Building Interoperable FHIR-based Vocabulary Mapping Services: A Case Study of OHDSI Vocabularies and Mappings. *Studies in health technology and informatics*, 245:1327, 2017.

[25] J. B. Kruskal and J. M. Landwehr. Icicle Plots: Better Displays for Hierarchical Clustering. *The American Statistician*, 37(2):162–168, May 1983. doi: 10.1080/00031305.1983.10482733

[26] T. v. Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. v. Wijk, J.-D. Fekete, and D. W. Fellner. Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. *Computer Graphics Forum*, 30(6):1719–1749, 2011. doi: 10.1111/j.1467-8659.2011.01898.x

[27] P.-M. Law, Z. Liu, S. Malik, and R. C. Basole. MAQUI: Interweaving Queries and Pattern Mining for Recursive Event Sequence Exploration. *IEEE transactions on visualization and computer graphics*, Aug. 2018. doi: 10.1109/TVCG.2018.2864886

[28] 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 *Proceedings of the 20th International Conference on Intelligent User Interfaces*, IUI '15, pp. 38–49. ACM, New York, NY, USA, 2015. event-place: Atlanta, Georgia, USA. doi: 10.1145/2678025.2701407

[29] S. Malik, B. Shneiderman, F. Du, C. Plaisant, and M. Bjarnadottir. High-Volume Hypothesis Testing: Systematic Exploration of Event Sequence Comparisons. *ACM Trans. Interact. Intell. Syst.*, 6(1):9:1–9:23, Mar. 2016. doi: 10.1145/2890478

[30] A. Mathisen and K. Grnbk. Clear Visual Separation of Temporal Event Sequences. In *2017 IEEE Visualization in Data Science (VDS)*, pp. 7–14, Oct. 2017. doi: 10.1109/VDS.2017.8573439

[31] T. E. Meyer, M. Monroe, C. Plaisant, R. Lan, K. Wongsuphasawat, T. S. Coster, S. Gold, J. Millstein, and B. Shneiderman. Visualizing patterns of drug prescriptions with EventFlow: A Pilot Study of Asthma Medications in the Military Health System. Technical Report HCIL 2013-13, 2013.

[32] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal Event Sequence Simplification. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2227–2236, Dec. 2013. doi: 10.1109/TVCG.2013.200

[33] S. N. Murphy, G. Weber, M. Mendis, V. Gainer, H. C. Chueh, S. Churchill, and I. Kohane. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association*, 17(2):124–130, Mar. 2010. doi: 10.1136/jamia.2009.000893

[34] C. Olston and E. H. Chi. ScentTrails: Integrating Browsing and Searching on the Web. *ACM Trans. Comput.-Hum. Interact.*, 10(3):177–197, Sept. 2003. doi: 10.1145/937549.937550

[35] C. Partl, S. Gratzl, M. Streit, A. M. Wassermann, H. Pfister, D. Schmalstieg, and A. Lex. Pathfinder: Visual Analysis of Paths in Graphs. *Computer Graphics Forum: Journal of the European Association for Computer Graphics*, 35(3):71–80, June 2016. doi: 10.1111/cgf.12883

[36] A. Perer and D. Gotz. Data-driven exploration of care plans for patients. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pp. 439–444. ACM, New York, NY, USA, 2013. doi: 10.1145/2468356.2468434

[37] A. Perer, I. Guy, E. Uziel, I. Ronen, and M. Jacovi. Visual social network analytics for relationship discovery in the enterprise. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 71–79, Oct. 2011. doi: 10.1109/VAST.2011.6102443

[38] A. Perer and F. Wang. Frequence: Interactive Mining and Visualization of Temporal Frequent Event Sequences. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, IUI '14, pp. 153–162.

ACM, New York, NY, USA, 2014. event-place: Haifa, Israel. doi: 10.1145/2557500.2557508

[39] W. Raghupathi and V. Raghupathi. Big data analytics in healthcare: promise and potential. *Health Information Science and Systems*, 2, Feb. 2014. doi: 10.1186/2047-2501-2-3

[40] S. Rea, J. Pathak, G. Savova, T. A. Oniki, L. Westberg, C. E. Beebe, C. Tao, C. G. Parker, P. J. Haug, S. M. Huff, and C. G. Chute. Building a robust, scalable and standards-driven infrastructure for secondary use of EHR data: The SHARPn project. *Journal of Biomedical Informatics*, Feb. 2012. doi: 10.1016/j.jbi.2012.01.009

[41] J. T. Rich, J. G. Neely, R. C. Paniello, C. C. J. Voelker, B. Nussenbaum, and E. W. Wang. A Practical Guide to Understanding Kaplan-Meier Curves. *Otolaryngology–head and neck surgery : official journal of American Academy of Otolaryngology-Head and Neck Surgery*, 143(3):331–336, Sept. 2010. doi: 10.1016/j.otohns.2010.05.007

[42] P. Riehmann, M. Hanfler, and B. Froehlich. Interactive Sankey diagrams. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pp. 233–240, Oct. 2005. doi: 10.1109/INFVIS.2005.1532152

[43] A. Rind. Interactive Information Visualization to Explore and Query Electronic Health Records. *Foundations and Trends in HumanComputer Interaction*, 5(3):207–298, 2013. doi: 10.1561/1100000039

[44] K. A. Spackman, K. E. Campbell, and R. A. Ct. SNOMED RT: a reference terminology for health care. *Proceedings of the AMIA Annual Fall Symposium*, pp. 640–644, 1997.

[45] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, pp. 57–65, Oct. 2000. doi: 10.1109/INFVIS.2000.885091

[46] C. Tominski, J. Abello, F. v. Ham, and H. Schumann. Fisheye Tree Views and Lenses for Graph Visualization. In *Tenth International Conference on Information Visualisation (IV'06)*, pp. 17–24, July 2006. doi: 10.1109/IV.2006.54

[47] N. TraCS. The North Carolina Translational and Clinical Sciences Institute Clinical Data Warehouse for Health. https://tracs.unc.edu/index.php/services/biomedical-informatics/cdw-h, 2015.

[48] A. Unger, N. Drager, M. Sips, and D. J. Lehmann. Understanding a Sequence of Sequences: Visual Exploration of Categorical States in Lake Sediment Cores. *IEEE transactions on visualization and computer graphics*, 24(1):66–76, 2018. doi: 10.1109/TVCG.2017.2744686

[49] F. van Ham and A. Perer. Search, Show Context, Expand on Demand: Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):953–960, 2009.

[50] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman. Aligning temporal data by sentinel events: discovering patterns in electronic health records. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pp. 457–466. ACM, New York, NY, USA, 2008. doi: 10.1145/1357054.1357129

[51] G. M. Weber, K. D. Mandl, and I. S. Kohane. Finding the Missing Link for Big Biomedical Data. *JAMA*, 311(24):2479–2480, June 2014. doi: 10.1001/jama.2014.4228

[52] W. Willett, J. Heer, and M. Agrawala. Scented widgets: improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1129–1136, 2007.

[53] K. Wongsuphasawat and D. Gotz. Outflow: Visualizing Patient Flow by Symptoms and Outcome. In *IEEE VisWeek Workshop on Visual Analytics in Healthcare*. Providence, Rhode Island, USA, 2011.

[54] K. Wongsuphasawat, J. A. Guerra Gmez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. LifeFlow: visualizing an overview of event sequences. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pp. 1747–1756. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979196

[55] K. Wongsuphasawat and J. Lin. Using visualizations to monitor changes and harvest insights from a global-scale logging infrastructure at Twitter. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 113–122, Oct. 2014. doi: 10.1109/VAST.2014.7042487

[56] F. Yates. Contingency Tables Involving Small Numbers and the 2 Test. *Supplement to the Journal of the Royal Statistical Society*, 1(2):217–235, 1934. doi: 10.2307/2983604

[57] Z. Zhang, D. Gotz, and A. Perer. Iterative cohort analysis and exploration. *Information Visualization*, 14(5), 2015. doi: 10.1177/1473871614526077

[58] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. MatrixWave: Visual Comparison of Event Sequence Data. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 259–268. ACM, New York, NY, USA, 2015. event-place: Seoul, Republic of Korea. doi: 10.1145/2702123.2702419

[59] W. Zhao, D. Borland, A. E. Chung, and D. Gotz. Visual Cohort Queries for High-Dimensional Data: A Design Study. In *Proceedings of Visual Analytics in Healthcare*. San Francisco, CA, 2018.

[60] M. Zinsmaier, U. Brandes, O. Deussen, and H. Strobelt. Interactive Level-of-Detail Rendering of Large Graphs. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2486–2495, Dec. 2012. doi: 10.1109/TVCG.2012.238