LifeFlow: Visualizing an Overview of Event Sequences

Krist Wongsuphasawat[1,2],
John Alexis Guerra Gómez[1,2],
Catherine Plaisant[2],
Taowei David Wang[1,2],
Ben Shneiderman[1,2]
Department of Computer Science[1] &
Human-Computer Interaction Lab[2]
University of Maryland, College Park, MD
{kristw,jguerrag,plaisant,tw7,ben}@cs.umd.edu

Meirav Taieb-Maimon[2,3]
Ben-Gurion University of the Negev[3]
Beer-Sheva, Israel
meiravta@bgu.ac.il

ABSTRACT

Event sequence analysis is an important task in many domains: medical researchers may study the patterns of transfers within the hospital for quality control; transportation experts may study accident response logs to identify best practices. In many cases they deal with thousands of records. While previous research has focused on searching and browsing, overview tasks are often overlooked. We introduce a novel interactive visual overview of event sequences called *LifeFlow*. LifeFlow is scalable, can summarize all possible sequences, and represents the temporal spacing of the events within sequences. Two case studies with healthcare and transportation domain experts are presented to illustrate the usefulness of LifeFlow. A user study with ten participants confirmed that after 15 minutes of training novice users were able to rapidly answer questions about the prevalence and temporal characteristics of sequences, find anomalies, and gain significant insight from the data.

Author Keywords

Timestamped Event Sequences, Temporal Categorical Data

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces—*Graphical user interfaces (GUI)*

General Terms

Design

INTRODUCTION

Event sequences are series of timestamped events. For example, Electronic Health Records (EHRs) contain doctor visits, lab results, medication orders, transfer among hospital departments etc. Highway incident logs, student records, web logs, usability study logs, are other examples.

Patient – (6:11, Arrive Hospital), (6:15, Emergency Room), (9:05, ICU), ...
Incident – (8:05, Incident Notification), (8:10, Police arrived), ...
Student – (28 Aug'07, Enter PhD program), (30 Apr'10, Proposal), ...

Event sequence analysis looks at (1) the sequence of events (e.g. Arrival→Intensive Care Unit (ICU)→Floor) and (2) time gaps between events (e.g. Emergency Room *(and after 5 hours)*→ Transfer to ICU). Medical researchers might be interested in analyzing transfer sequences in hospitals to improve the quality of care, while traffic managers are interested in analyzing incident logs to identify best practices.

Previous work on temporal data visualization can support many types of analysis, ranging from examining a single record in details to various ways of filtering and searching multiple records. They can answer questions regarding the number of records that include a specific event sequence but questions requiring an overview are not adequately supported. E.g., a question such as "What are the most common transfer patterns between services within the hospital?" requires examination of all records one by one. Being unable to see all records on the screen at once makes it difficult to spot any pattern. Squeezing a billion records into a million pixels [30] is a great challenge in information visualization. In this paper we introduce a novel interactive visual overview of event sequences called *LifeFlow*, which is scalable and can summarize not only all possible sequences but also the temporal spacing of the events within sequences. We first describe a motivating example from the medical domain, then introduce the LifeFlow visualization, review related work, describe the user interface interactive features, and finally present the results of our evaluations (two case studies and a laboratory study with ten participants). While the examples in this paper focus on the medical domain, the technique is widely applicable to other fields, such as incident management, log analysis, or the study of human activities in general. To gain a further understanding of the interactive experience, see the videos at http://www.cs.umd.edu/hcil/lifeflow.

MOTIVATING CASE STUDY

The use of information visualization to support patient care and clinical research is gaining momentum [1, 19, 9]. We have been working with physicians from the Washington Hospital Center for several years, and continuously identifying case studies that inform the design of our tools and help us

evaluate the usefulness of our approaches. This section describes a particular case study that motivated the original design of LifeFlow. It was conducted with Dr. Phuong Ho, a practicing physician in the emergency department, who was interested in analyzing sequences of patient transfers between departments for quality assurance.

Event Definitions

These terms will be used when describing the case study.
1) *ER*: Emergency Room, the section of a hospital intended to provide treatment for victims of sudden illness or trauma
2) *ICU*: Intensive Care Unit, a hospital unit in which patients requiring close monitoring and intensive care are kept
3) *IMC*: Intermediate Care, a level of medical care in a hospital that is intermediate between ICU and Floor
4) *Floor*: a hospital ward where patients receive normal care

Example question

One of Dr. Ho's particular interests was the monitoring of *bounce backs*, which occurred when a patients' level of care is decreased then increased back again urgently. E.g., a patient's condition might have improved enough to have him transferred from the ICU to Floor, but his condition worsened again and he had to be sent back to intensive care within 48 hours, suggesting he might have left the ICU too early. This pattern corresponds to a hospital quality metric that is very rarely monitored. Dr. Ho had been using an MS Excel spreadsheet to find these patients. In an interview he described the complex and time consuming effort to create the formulas and view the data. This is due in part to the fact that there are many room types and special conditions for using those rooms. We had previously worked with Dr. Ho using *LifeLines2* [35] and Similan [39] to locate patients with specific known event sequences such as the one described above. Once it had become easy to search for specific known sequences we identified other questions that could not be answered as easily; e.g.: what typically happens to patients after they leave the ER, or the ICU; what are the most common transfer patterns, what is the percentage of patients transferred from ICU to Floor and how long does it take, are there any unexpected sequences? All those new questions require a summary of all the transfer sequences and their temporal attributes. To this end, we propose *LifeFlow* to provide an overview of all transfer sequences.

LIFEFLOW DISPLAY

Figure 1 illustrates the conversion from four records of event sequences to the LifeFlow display. Raw data are displayed on a horizontal timeline with colored triangles representing events (in the same approach as LifeLines2 [35]). Each row represents one record. The transformation into LifeFlow takes two steps: aggregation and visualization.

1) *Aggregation* – All records are aggregated into a tree-based data structure called a *tree of sequences* based on the prefixes of their event sequences. For example, a record that contains event sequence Arrival → ER → ICU and a record that contains event sequence Arrival → ER → Floor, share the same prefix sequence Arrival → ER. The records are grouped event-by-event from the beginning of the event sequences to the end. In Figure 1, all records start with the

Figure 1. This diagram explains how a LifeFlow display can be constructed to summarize four records of event sequences. Raw data are represented as colored triangles on a horizontal timeline (using the traditional approach also used in LifeLines2). Each row represents one record. The records are aggregated by sequence into a data structure called a tree of sequences. The tree of sequences is then converted into LifeFlow visualization. Each tree node is represented with an event bar. The height of the bar is proportional to the number of records while its horizontal position is determined by the average time between events.

Figure 2. The Icicle Tree Visualization - the inspiration for Lifeflow - is used to display hierarchical information. The biggest rectangle on the top represents the root node. The children nodes are placed under their parent nodes. Each node's width is determined by a sum of its children's width.

blue event so they are grouped together (indicated by dashed rectangle) in to a blue tree node. Then, they all also have the pink event, so they are still grouped together into a pink node. In the next step, two of them have red events while the other two have green events so they are split into red and green nodes. Then do the same for the rest of the event sequences. In some situations, users may chose to group consecutive events of the same type together when building the tree of sequences. For example, two consecutive transfers to Floor can be treated as one transfer with the second transfer ignored. A record with sequence Arrival → ER → Floor → Floor → ICU is treated as Arrival → ER → Floor → ICU. This aggregation option can be turned on or off as needed.

2) *Visualization* – Once the tree of sequences is created, it can be visualized into the LifeFlow display. Each node of the tree is represented with a color-coded *event bar*, matching the color of the event type. The height of a bar is determined by the number of records in that node proportionally to the total number of records. E.g., the red node contains two out of four records so the height of its corresponding event bar is 50% of the total height. The horizontal gap between a bar (e.g. pink bar) and its parent (the blue bar on its left) is proportional to the mean time between the two events (blue → pink). By default, the representative time gap is the mean, but users can select other metrics, such as the median.

The LifeFlow display is scalable because it does not require additional space for a larger number of records. Users can display hundreds of records or millions of records using the same amount of screen space.

RELATED WORK
Temporal Data Visualization
Single-Record Visualization – There has been a long history of visualizing event sequences. Many systems were designed for analyzing a single record [10, 14, 18, 26, 4, 2]. The most common approach is to use a timeline-based representation. Events are placed on a horizontal timeline according to their time. One record consists of multiple rows, one for each category of events. Spiral timelines – in which angle represents time interval (time of day, days of week, months, or years)– were inspired by the cyclic nature of how we organize time and used to reveal periodic patterns in time series [8, 16, 37].

Visualization of multiple records in parallel – To support the analysis of multiple records, a common technique is to stack instances of single-record visualizations. [11, 33]. Weber *et al.*'s spiral-based visualization [37] also supports multiple records by showing each record as one ring. Those tools typically provide searching and filtering mechanisms [35, 39, 34]. In the context of our case study, users of those tools could find patients who were transferred with a specific known sequence, or find patients who were admitted to the ICU at least once, but could not find out what the common sequences are or spot anomalous sequences. Some tools allowed the users to organize records into hierarchy [7] or groups [25]. Timeline Tree [7] is a tree with timelines at the leaf nodes. Progressive Multiples [25] allow users to manually place multiple timelines in folders. However, unlike LifeFlow which provides one visual abstraction that represents multiple timelines, they do not provide any abstraction.

Visualization of combined records – To overcome the scalability issue and provide an overview, some systems provide an abstraction of multiple records [15, 3, 36]. *Continuum* [3] shows a histogram of frequency of events over time while *LifeLines2* [36] has a *temporal summary*, which is a stacked bar chart that shows the distribution of event types within each period of time. These methods can provide the distribution of events by time, which answers the questions related to the distribution, such as which type of event occurred most frequently in January 2007 or which type of event usually occurred within the first week after patients arrived at the hospital? However, the event sequences within the records are obscured and thus, it cannot answer questions related to sequences, such as where did the patient usually go directly after arrival or what is the most common transfer sequence?

Hierarchy Visualization
In LifeFlow, all records are first grouped into a hierarchical structure called a tree of sequences. Many visualizations can display hierarchical structures [32]. The most common way is to display a node-link layout in 2D [5, 38, 27], or 3D [28] or hyperbolic space [21, 24]. Space-filling techniques use implicit containment and geometry features to present a hierarchy [17, 20, 12, 6, 32]. *Icicle tree* [20, 12, 6], also called *Icicle plot*, displays hierarchical data as stacked rectangles, usually ordered from top to bottom. This visualization directly inspired our LifeFlow design (Figure 2). The root takes the entire width. Each child node is placed under its parent with the width proportional to the percentage it consumes relative to its siblings. Using these methods, the sequences of events can be represented. However, the length of time between events – which is important in many analyses – is not represented. A *phylogenetic tree* [22] is a branching diagram showing the inferred evolutionary relationships among various biological species. The edge lengths in some phylogenetic trees may be interpreted as time estimates, but each node in the tree represents only one species while each node in LifeFlow represents multiple records of event sequences. *VizTree* [23] uses a tree-based representation to detect patterns within a very long event sequences (like EKGs). For example, for a record that contains sequence of events "A,B,A,B,B,B", VizTree creates a prefix tree of all subsequences (ABA, BAB, ABB, ...), then displays the prefix tree using edge thickness to represent frequency, which can reveal frequent and rare subsequences. As with other tree-based approaches, VizTree also ignores the duration of the time gap between events. Also, VizTree focused on analyzing event sequences of length n in a single very long record rather than providing an overview of millions of records.

In summary, LifeFlow represents a new and powerful approach to aggregate event sequences, much like a compact Table of Contents is a powerful addition when there is only a search box for a full length book. LifeFlow allows users to see not only all possible sequences but also their prevalence and summarize information about the time gap between events, which is not supported by other visualizations.

LIFEFLOW INTERACTIVE FEATURES
Our working prototype of LifeFlow (Figure 3) includes the following interaction features to support exploration:
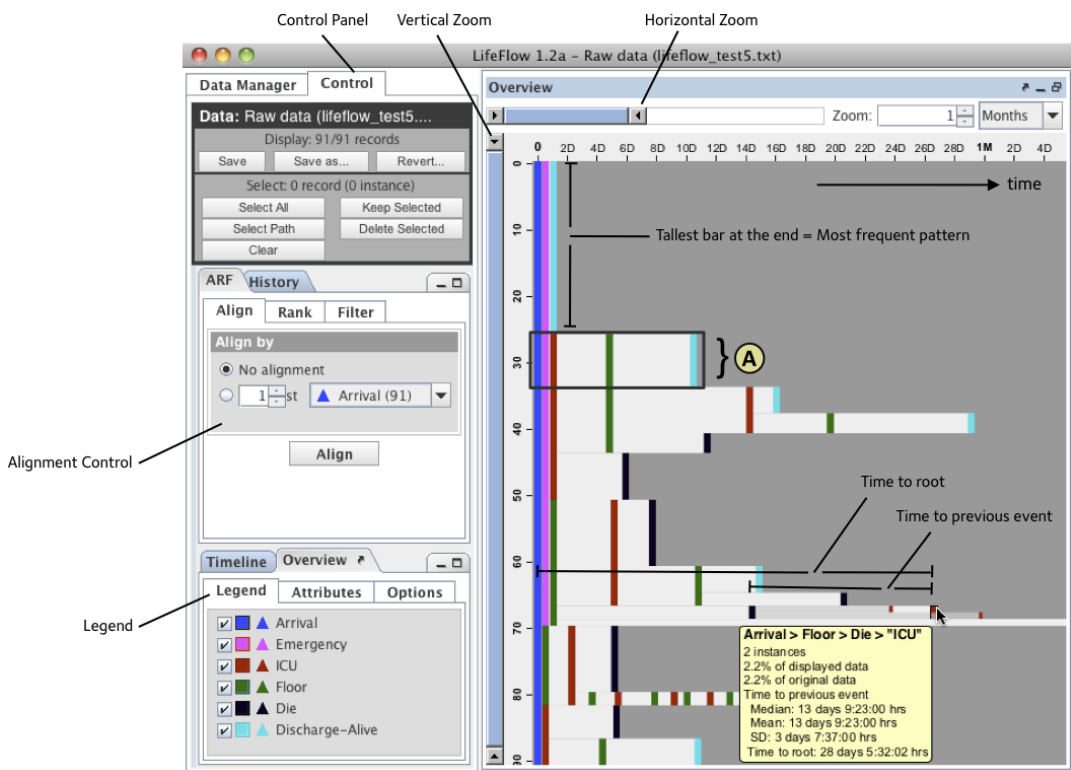
Figure 3. This screenshot of LifeFlow shows a random sample of patient transfer data based on real de-identified data (as used in the user study). The way to read sequences in LifeFlow is to read the colors (using the legend). For example the sequence (A) in the figure is Arrival (blue), Emergency (pink), ICU (red), Floor (green) and Discharge-Alive (light blue). The horizontal gap between colored bars represents the average time between events, showing that it takes approximately 4 days on average for patients to go from ICU to the floor in that (A) sequence . The height of the bars is proportional to the number of records, therefore showing the relative frequency of that sequence. The bars (e.g. Floor and Die) with same parent (Arrival → Emergency → ICU) are ordered by frequency (tallest bar on top), as you can see that Floor (green bar) is placed above Die (black bar). To find the most frequent pattern is to find the tallest bar at the end. Here it shows that the most common sequence is Arrival, Emergency then Discharge alive. Surprisingly, two patients were reported dead before transferred to ICU which indicates a data entry problem. (Please note that this is only a sample dataset and does not reflect the real performance of any hospital.)
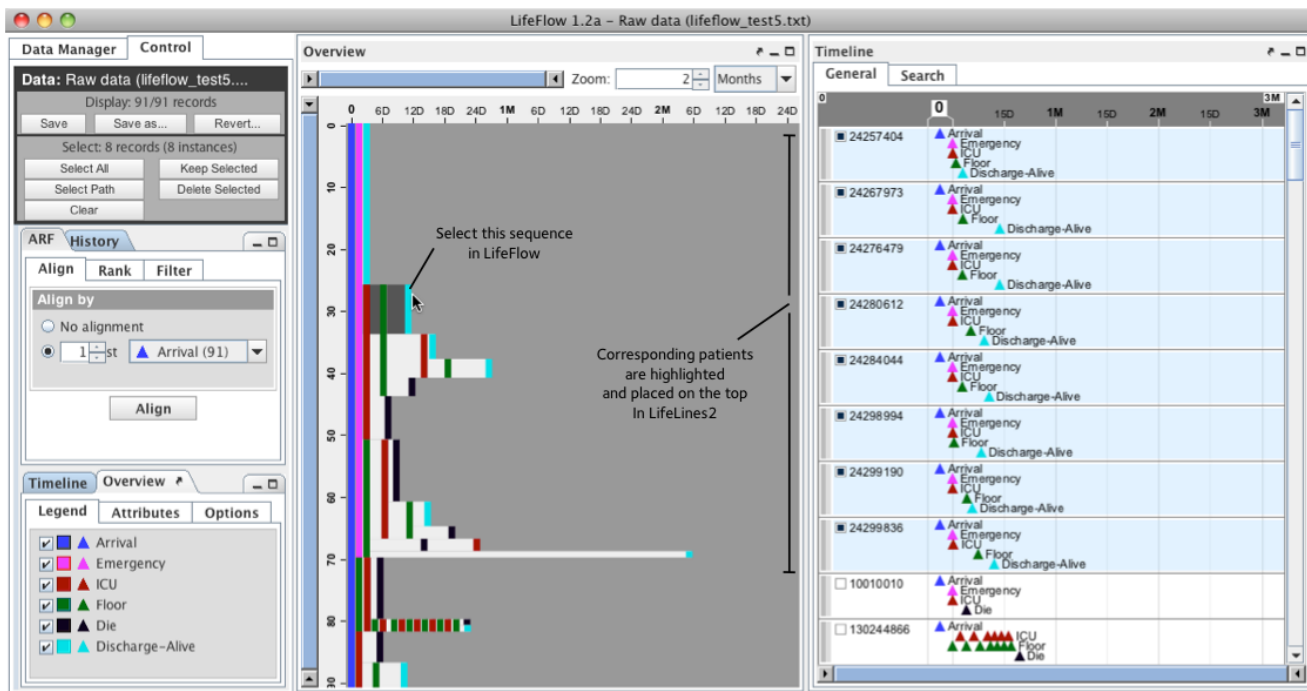
Figure 4. Here LifeFlow is used side-by-side with LifeLines2 so that individual records can be reviewed by scrolling. When a user clicks on a sequence in LifeFlow, the sequence is highlighted and all corresponding records are also highlighted and moved to the top in LifeLines2, allowing the user to examine them in more details.
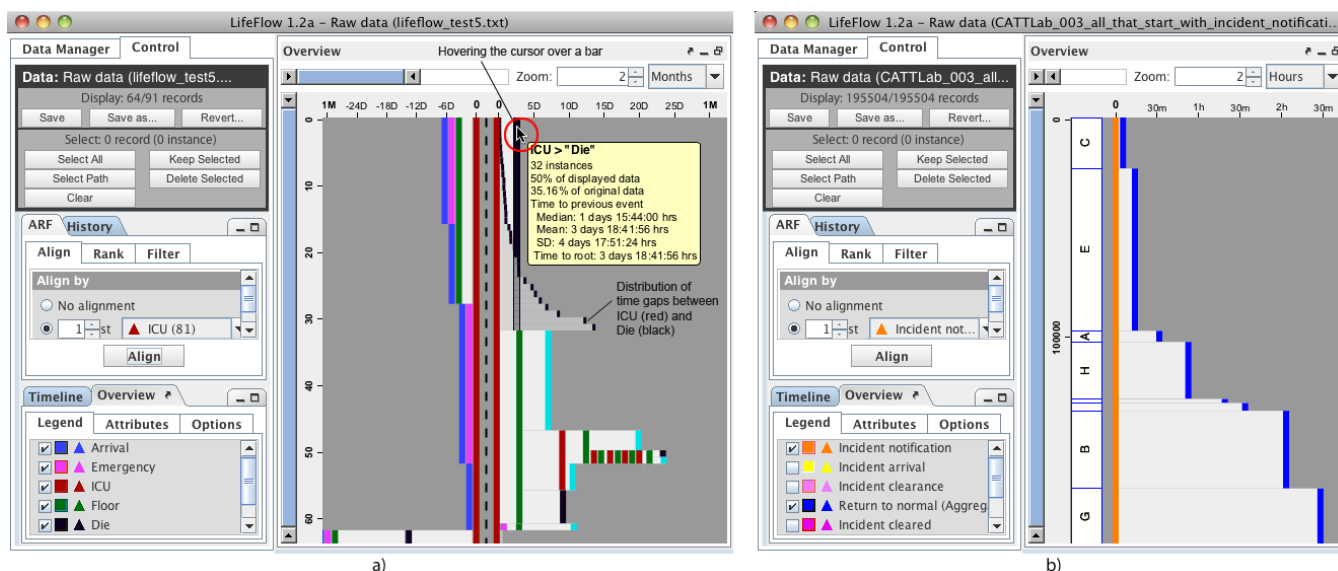
Figure 5. a) LifeFlow with alignment – The same data with Figure 3 was aligned by ICU. The user can see that the patients were most likely to die after a transfer to the ICU than any other sequence. (Please note that this is a random sample dataset and does not reflect the real performance of any hospital.) b) LifeFlow with traffic incidents data – The incidents are separated by agencies (A-G). Only "Incident Notification" and "Return to normal (aggregated)" events are shown. Other events are hidden. The agencies are sorted by a simple measure of agency performance (average time from the beginning to the end). Agency C seems to be the fastest to clear its incidents, followed by E, A, H, D, F, B and finally G.

1) *Zooming* – The horizontal zoom changes time granularity while the vertical zoom allows the users to see rare sequences in more detail.

2) *Tooltip* – When users move the mouse cursor over an event bar (Figure 3), a tooltip displays the full sequence of events, and some statistical information, such as mean time between events, standard deviation, etc.

3) *Overlay distribution of gap between events* – Hovering the cursor over a bar displays the distribution of time gaps overlaid on the Lifeflow. Figure 5a shows the distribution of length of stay in the ICU before the patients died.

4) *Sort* – Users can sort the sequences with the same parent in different ways: by the number of records that the bars represent (tallest bar on top) (Figure 3) or by the average time to previous event (shortest time on top) (Figure 5b). The default is to sort by number of records.

5) *Integration with LifeLines2* – LifeFlow can function as a standalone tool but combining with LifeLines2 facilitates exploration by allowing users to review individual records as details on demand [29]. By clicking on any event bar, users select all records that are included in that bar (Figure 4). Selected records are highlighted in the LifeLines2 view. Users can then choose to keep only the selection and remove everything else, or vice versa. In a symmetrical fashion, selecting a record in the LifeLines2 view highlights the pattern contained in that record in the LifeFlow view, allowing the users to find other records that contain the same sequence.

6) *Align* – Inspired by LifeLines2, LifeFlow allows users to choose any event type to be the *alignment* point. This supports tasks such as "what happened to the patients before and after they went to the ICU?" By default, the alignment point is not specified, so all records are aligned by the first event in the record. When users chose an alignment event, two

trees of sequences are built: one tree for the sequences before the alignment (from right to left) and another tree for the sequences after the alignment (from left to right). Figure 5a shows LifeFlow with alignment. The vertical dashed line marks the aligned event. The left and right side are what happened before and after the reference point, respectively. Now one can see that in this dataset, patients most often come to ICU from Floor. After that, they often died.

7) *Including non-temporal attributes* – Records usually also contain non-temporal attributes, e.g., patient's gender. While LifeFlow does not focus on displaying those attributes, it allows users to select a non-temporal attribute and groups records by that attribute before the sequences are aggregated. LifeFlow in Figure 5b groups traffic incident records by agency before aggregating by sequence, therefore allowing simple comparison between agencies.

8) *Include/Exclude event types* – Using the legend on the left side of the screen users can check or uncheck event types to include or exclude them from the sequences. This simple functionality allows powerful transformations of the display to answer questions. E.g., in Figure 5b, the user unchecked all event types except "Incident Notification" and "Return to normal", i.e. the beginning and end of incidents. All other events that could occur during an incident are ignored and LifeFlow regenerated to show only those two event types, allowing rapid comparisons between the agencies in terms of number of incidents and average time to clear the incidents.

9) *Displaying all event bars with equal height* – When data includes a large number of sequences, it could be difficult to review the rare sequences because they are represented with very thin bars. This option displays all leaf nodes using equal height regardless of the number of records, makes it easier to review and select rare sequences.

EVALUATION

We reported two case studies in two different domains following the Multi-Dimensional In-Depth Long-Term Case Studies (MILC) [31] approach. The two studies were done in collaboration with the physicians at the Washington Hospital Center and the Center for Advanced Transportation Technology Laboratory at the University of Maryland, respectively. In addition, we conducted a user study with 10 participants.

CASE STUDY I: MEDICAL DOMAIN

As discussed in the beginning of the paper, the design of LifeFlow was motivated by the patient transfer case study. As we developed LifeFlow we continued to work with Dr. Phuong Ho to analyze more patient transfer data, and in more details. We had a 1-2 hours meeting with him approximately every two weeks for 3 months. Before the meeting he provided us with the data that he wanted to analyze, and a few initial questions. We converted the data and during the meeting sat down and looked at the data together. After discussing the questions sent in advance, he would come up with additional questions and gave feedback about the user interface, therefore closely guiding our development. We summarized the results as follows:

First impression – The first time we showed LifeFlow to Dr. Ho using patient transfer data, he was immediately enthusiastic and confident that the tool would be useful for looking at all patients who came to the hospital and in particular the emergency room (ER). He knew that many people would want to see the typical flow of the patients and the transfer time between rooms. In another meeting, we received additional feedback from the director of the Emergency Department. Finding the bounce back patients visually in the display elicited comments such as "Oh! This is very cool!" and led to a discussion of the possibilities of using this tool to analyze hospital data in the long run.

Understanding the big picture – One of the datasets we received included all 7,041 patients who came to the ER in Jan 2010. Each record contains room assignments, time that the patient was assigned to each room, and how he/she was discharged from the hospital: dead, alive, leave without being seen (LWBS) and absence without leave (AWOL). We preprocessed the data by grouping the room numbers (e.g. ER15-P, 2G06-P) into types of room (ER, ICU, IMC, Floor). In the meeting, we showed LifeFlow and Dr. Ho could review the flow of patients in the hospital. The first thing that he noticed was the most common pattern, Arrival → ER → Discharge-Alive. 4,591 (65.20%) of the patients were not admitted to the hospital (discharged directly from the ER). This is regular and consistent with what he had expected because most of the patients who visited the ER were not in severe condition and could leave immediately after they received their treatment, so we removed these 4,591 patients from the visualization to analyze other patterns. The second most common pattern, Arrival → ER → Floor → Discharge-Alive (1,016 patients, 14.43%), now became more obvious. We decided to remove it too because it was also regular. We followed the same strategy, selecting regular common sequences and removing them from the visualization to detect the uncommon cases that might be irregular. Our partner noticed that 193 patients (2.74%) left without being seen while 38 patients (0.54%) were absence without leave. These two numbers could be compared with the hospital standard for quality control. Then, our partner saw two patterns that he was interested in (Arrival → ER → Floor → IMC and Arrival → ER → Floor → ICU). These patterns correspond to another quality control metric called *step ups*, which occurs when the patients were admitted to a lower level of care (Floor), but later transferred to a higher level of care (IMC or ICU). Dr. Ho could quickly see from the visualization that the patients were transferred from Floor to ICU faster than Floor to IMC on average so he used the tooltip to see the distribution. He captured screenshots to compare with practices reported in the research literature, but also commented that the average time seemed quite good from his knowledge.

We also demonstrated the alignment and used it to analyze the transfer flow before and after the patients were admitted to the ICU. From the total 181 ICU patients, 85 (46.96%) of them were transferred from the ER and 119 (66.75%) of them were transferred to the Floor after that. However, 6 patients were transferred back from Floor to ICU (bounce backs). We saw from the distribution that one patient was transferred back in less than a day. Dr. Ho requested to see these 6 patients in more details so we clicked on the bar, which highlighted these patients in LifeLines2 view and noted down these patients' ID. In addition, he also noticed some anomalous sequences, e.g. a few patient records showed a transfer to ICU after being discharged dead, pointing to obvious error in data entry or at least reflecting the possible delays in data entry. Although we did not identify other surprising transfers (which on the other hand, reflected good quality of care), this still showed that the tool is useful for monitoring the patient transfer flow. We also received additional questions from Dr. Ho after the meeting. Some questions included clear request for alignments ("I want to see this in LifeFlow. Specifically I want to see a view that shows me where all my [ICU] patients are coming from.") indicating that he could understand what the tool is capable of and how to use it for his task.

Measuring the transfer time – Because LifeFlow can easily calculate an average time, Dr. Ho formulated many queries asking about average time, such as "Of patients who came to the ICU from the ER, what was the average time it took for transfer of the patient to the ICU? More specifically, if they went to the ICU, how long did it take from the time they had arrived at the ER to get to the ICU? Same question for IMC..." or "For all the quarters, Jan-Mar 09, Apr-Jun 09, Jul-sep 09, Oct- Dec 09 and Jan-Mar 10, I want average time from ER to 2G [which is a specific type of ICU room]."

Comparison – Another use of LifeFlow was to compare different data sets by inspecting the difference between two side-by-side LifeFlow visualizations. Dr. Ho had a hypothesis about whether IMC patients were transferred faster during the day (7am-7pm) than during the night. We opened the same dataset in two windows and filtered the records by time-of-day filtering, making the two windows contain only patients who arrived during the day and during the night, respectively. We inspected the difference between the two visualizations but no significant difference was found. In an-

other case, we compared patients who were admitted to the IMC at least once in four quarters: Jan-Mar, Apr-Jun, Jul-Sep and Oct-Dec 2008. We opened the four datasets in four LifeFlow windows and noticed a difference in the patients who were transferred from ER to the ICU. In the first, third and fourth quarter, these patients were later transferred to IMC and Floor, with majority were transferred to the IMC. However, in the second quarter, all patients were later transferred to the IMC, suggesting further investigation whether this occurred by chance or any particular reason.

Discussion

We found that our domain expert was able to understand LifeFlow rapidly and that LifeFlow was useful to provide an overview of the entire data set, and to compare and measure the transfer times. Once the data was loaded, he could quickly see the big picture and find anomalies. Dr. Ho expressed that being able to formulate queries easily gave him more time to look at the data and formulate new hypotheses or think about other interesting questions. Although he might have been able to answer of some of the questions in SQL, it would be very difficult and error prone. He also mentioned that LifeFlow would be very useful for long-term monitoring and quality control because it provides a quick way to inspect the data from the overview. Another researcher who recently started a case study with us also gave this comment on LifeFlow: "Statistical techniques for dealing with longitudinal data generally focus on changes in continuous variables over time, and the problem of identifying patterns of sequence and temporal spacing in categorical events is not handled by standard techniques and software. This problem arises a lot in analysis of health care data, and this tool opens up a kind of study that just hasn't been possible before."

CASE STUDY II: TRANSPORTATION DOMAIN

To illustrate how LifeFlow is not in anyway limited to medical applications, we describe a second case study currently underway with the Center for Advanced Transportation Technology Lab (CATT Lab) at the University of Maryland [13]. We are using LifeFlow to examine a data set from the National Cooperative Highway Research Program (NCHRP) that includes 203,214 traffic incidents from 8 agencies. Each incident record includes a sequence of incident management events: *Incident notification* –when the agency is first notified of the incident, *Incident Arrival* –when the emergency team arrives the scene, *Lane Clearance* –when the lanes are opened, but the incident scene may be not completely cleared, *Incident cleared, Incident clearance*, and *Return to normal* –all denote the end of incidents. For ease of analysis, we aggregated all three into the new event type *Return to normal (aggregated)*. A typical sequence should start with "Incident Notification" and finish with "Return to normal (aggregated)", with the possibility of having "Incident Arrival" and "Lance Clearance" in between. In addition, the traffic incidents data includes two attributes: the agency (represented with a letter from A to H for anonymity) and the type of incident (e.g. Disabled Vehicle, Fatal Accident, etc.)

Quantifying data quality issues – After loading the dataset in LifeFlow, we immediately noticed that the Agency B con-

tains a sequence that was more than 110 years long with 6,712 incidents. Investigating further, we found that Agency B reported the "Incident Arrival" of those incidents as January 1th 1900. Since this date is commonly used as the initial date in computer systems, this suggested that the system the Agency used to register this event might have used it as a default value. Considering these incidents as corrupted data, we removed all of them from the dataset. While it was easy to spot this problem, such anomalies can often remain undetected, and skew the results of even the simplest of analysis such as calculating the mean time to clearance. Similarly, we found and removed 48 incidents from Agency D that are about 10 months long, in which the "Incident Arrival" occurs before the "Incident Notification".

The next thing we noticed from the data was that there were many incidents that lasted exactly 24 hours, which seemed unlikely. Using the Align, Rank and Filter operators, we found that those 24-hours-long incidents have "Incident arrival" events occur in the first hour of the day (e.g. 12:30AM April 10 2009) and "Incident notification" events happened in the last hour of the same day (e.g. 11:50PM April 10 2009). This observation seemed to suggest that there were data entry problems with those incidents, indicating that the operator failed to – or was not able to – record the correct date of an event (e.g. 12:30AM Apr 11, 2009 as opposed to 12:30AM Apr 10, 2009). Similar errors were discovered for paths that are about 12 hours long, in which case the errors seem to be problems choosing between AM and PM in the date. Those anomalies were found quite easily by the computer scientist developer, who had no experience in transportation data. Finding such errors using traditional tools like SQL or manual analysis can be very difficult and time consuming, and requires experienced analysts who would suspect the existence of such errors.

Ranking the agencies' performance – In this study, we used the time from when the agencies were notified to the final clearance of the incidents as a performance measure. The time when the agency was notified can be indicated by the "Incident Notification" event. In order to compare the agencies performance, we first removed the inconsistent data (incidents that do not start with "Incident Notification"), which could be performed easily using the equal height overview feature. After the steps above, the visualization of the data can be seen in Figure 5b. Incidents are grouped by agencies. We showed only two event types ("Incident Notification" (orange) and "Return to Normal (Aggregated)" (blue)), so the horizontal length of each agency's path represents the average time from incident notification to final clearance, which reflects the performance measure for that agency. We then sorted the agencies according to the length of their paths, resulting in the fastest agency (shortest path) on the top and the slowest agency (longest path) in the bottom. From Figure 5b we could see that Agency C was the fastest agency to clear its incidents, taking about 5 minutes in average, while the slowest one was Agency G with an average of about 2 hours 27 minutes. To investigate deeper into Agency C's data, we removed the data from other agencies and looked into the different incident types reported. We found that most of the incidents that Agency C reported are "Disabled Vehicles"

which had in average a time to clear of about 1 minute. Looking at the event distribution, we also found that a large number of the incidents reported "Clearance" immediately after "Incident Notification". This observation made us wonder if there is any explanation for these immediate clearances, and encouraged further analysis. In a similar fashion, we investigated Agency G, which seemed to be the slowest agency. Agency G classified their incidents in only two types "Non-ATMS Route Incident" and simply "Incident". The "Incident" incidents had an average length of about 38 minutes, which is a very good time compared to the other agencies. However, the "Non-ATMS Route Incident" incidents took in average 5 hours 14 minutes to clear. So we realized that when using the average time of all incidents from Agency G without considering the incident types, Agency G seemed to be slower than other agencies. While in fact, Agency G performed quite well for incident type "Incident".

Discussion

Although our data analysis in the case study was limited and preliminary, domain experts from the CATT Lab are conducting a more formal analysis of the data. They reviewed our work and stated that they wished LifeFlow was available earlier on when they started their own analysis. They confirmed the existence of anomalies that we had found in the data, and stated that their elimination was non-trivial when using SQL because they had to expect the errors in advance and be careful to exclude them from their analysis. However excluding all the possible erroneous sequences in a SQL query would be very difficult. In the end, they needed to review the results of SQL queries to ascertain that there were no longer any errors. Without LifeFlow, this kind of review and identification of unexpected sequences would be almost impossible. Finally, they mentioned that LifeFlow would allow them to ask more questions faster, and probably richer questions about the data. LifeFlow was also able to reveal unexpected sequences that may have been overlooked, but the tool also suggested that their prevalence is limited. We believe that using LifeFlow can assist analysts explore large datasets, such as the NCHRP traffic incident information, in ways that would be very difficult using traditional tools and might allow analysts to find richer results in less time.

USER STUDY

Our goal in this study was to investigate if LifeFlow was easy to learn, and if users could use the interface efficiently to answer representative questions. We also wanted to observe what strategies users chose and what problems they would encounter, and gather feedback and suggestions for further improvement. We used a dataset that included 91 records of hospital patient transfer (a subset of our real de-identified data, which included known anomalies to see if participants could find them). Because medical professionals have very little availability, are hard to recruit for a user study, and the data used in the study is simple enough to be understood by students, the participants we used for this study were graduate students (5 male and 5 female) from various departments of the University of Maryland. None of them was a member of the LifeFlow development team.

Procedure, Tasks and Results

Training consisted of a 12-minute video and five training questions. When the participants could answer the questions correctly, they could start the study tasks. The order of the tasks was randomly permuted across participants. The tasks were representative of the questions proposed by our domain experts during the case study, and designed to test the usability of the main interaction techniques of LifeFlow. We encouraged participants to think aloud while performing the tasks. For the first 14 tasks, observers recorded completion time and errors, if any. Because the participants needed time to understand the tasks, we gave them time to read the task description before starting the timer.

Tasks 1-9: Simple Features – The first 9 tasks required understanding the LifeFlow display and using simple interactive features such as tooltips or zooming. The tasks included questions about frequent sequences (e.g. "Where usually did the patient go after they arrived?", "What is the most common pattern?"), number of records in specified sequences (e.g. "How many patients went from arrival directly into ICU?"), time between events (e.g. "How long is the average time from ER to ICU?") and comparison between sequences (e.g. "After arriving to the ER, patients might be transferred to Floor or the ICU. Were the patients transferred from the ER to the ICU faster than from the ER to the Floor?").

Task 10-14: Advanced Features – Four tasks required using advanced features, such as alignment or using LifeFlow and LifeLines2 in combination (e.g. "Usually, where were the patients before they were admitted to the ICU for the first time?", "Retrieve the IDs of all patients with this transfer pattern." or "How many patients had the same transfer pattern as patient no.10010010?").

Results for tasks 1-14: The participants were able to perform the simple and advanced tasks quickly. They were able to use the interactions to adjust the visualization and retrieve information that was not presented in the initial view. The average ± SD completion time for the simple and advanced tasks were 14.9 ± 12.7 seconds and 15.8 ± 12.5 seconds, respectively. Please note that the participants were also narrating their actions while performing the tasks, which might have slowed them down. Only one participant made one mistake in a complex task: while retrieving the IDs of all patients who were transferred with a particularly long sequence she misread a color and could not find the correct sequence. However, she knew what she needed to do to retrieve the IDs after the sequence was found.

Task 15: Overall analysis and finding anomalies – In the last task, we asked the participants to imagine themselves as a manager who was trying to evaluate the performance of the hospital. We gave them 10 minutes to find any surprising, exceptional or impossible sequences that might indicate a problem in data entry or in hospital procedures, and explain why they thought it was a problem. We told them to report as many insights as they could in 10 minutes. We had planted three (realistic) data anomalies: 1) A few patients who died before being transferred to ICU. 2) Patients who bounce back and forth between the ICU and Floor several times. 3) Patients who stayed in the Floor for 3 months.

Results: Eight out of ten participants were able to detect all three anomalies. Two participants reported only the first two anomalies, but when we directed their attention towards the third anomaly, they explained that they had noticed it but did not think it was abnormal because some patients (e.g., cancer patients) might stay in the hospital for a long time. In addition, they also provided insight about other sequences that were possible, but undesirable from a manager's perspective, such as there were many patients who died in the Floor. They also reported surprising patterns, such as many patients were discharged alive directly from the ICU. We also observed the following common strategies: 1) Examine things that catch the eye first. 2) Scan all sequences systematically top to bottom – When they saw the normal sequence, e.g., ICU to Floor, they noted that this is good and moved on to the next sequence. 3) Align by each type of event 4) Hovering over bars with the mouse to explore distribution and detect outliers from the distribution 5) Check for more details in LifeLines2 – Although we did not display LifeLines2 at the beginning of this task, several participants opened it to use the combined view. All participants used strategies 1-3 consecutively. Three participants also followed with strategy 4. Four participants followed with strategy 5.

Debriefing: During the debrief, typical comments included: "The tool is easy to understand and easy to use.", "very easy to find common trends and uncommon sequences", "The alignment is very useful.", "In LifeFlow, it is easier to see the high level picture. With LifeLines2, you can check individuals. LifeFlow provides a great summary of the big picture." Common suggestions for improvement included increasing the bar width to make it easier to select and reorganizing the tooltip to make it more readable. Two participants also asked to analyze the data from their own research with LifeFlow.

Summary

Results suggest that users can learn to use LifeFlow in a short period of time and that LifeFlow's overview of the data allows them to understand patterns and find anomalies. There were several common strategies used when performing data analysis but not every participant used all strategies, which indicated the need for a framework to support data analysis.

CONCLUSIONS AND FUTURE WORK

Analyzing large numbers of event sequences is an important and challenging task. We introduced a new scalable visualization called LifeFlow that provides an overview of event sequences to support users' exploration, and reported on two case studies and one user study. One case study was conducted with the physicians whose questions motivated our research. The feedback from the physician was very positive. They could see clearly how this tool could be used for quality assurance in the hospital. Also, our physician partner spent less time figuring out how to specify queries and more time thinking about other interesting questions. Another case study was conducted with traffic incidents data to demonstrate the generality of the approach goes well beyond the medical domain. We were able to eliminate many errors in the data and compare different agencies' performance, thereby generating positive feedback from the do-

main experts. Finally, we report on a user study with ten participants which confirmed that even novice users with 15 minutes of training were able to learn to use LifeFlow and rapidly answer questions about the prevalence of interesting sequences, find anomalies, and gain significant insight from the data. We believe that LifeFlow can be applied to many other fields, where event sequences are the main focus, such as student progress analysis, usability study or web log analysis, and human activities log analysis in general. The next step of our research will focus on creating a framework to guide the data analysis, supporting comparison between datasets, integration with searching and filtering and including new interaction techniques.

REFERENCES

1. Aigner, W., and Miksch, S. CareVis: integrated visualization of computerized protocols and temporal patient data. *Artificial intelligence in medicine 37*, 3 (2006), 203–18.

2. Aigner, W., Miksch, S., Thurnher, B., and Biffl, S., PlanningLines: Novel Glyphs for Representing Temporal Uncertainties and Their Evaluation. *Proc. Int'l Conf. Information Visualisation* (2005), 457–463.

3. André, P., Wilson, M. L., Russell, A., Smith, D. A., Owens, A., and Schraefel, M., Continuum: designing timelines for hierarchies, relationships and scale. *Proc. ACM UIST* (2007), 101.

4. Bade, R., Schlechtweg, S., and Miksch, S., Connecting time-oriented data and information to a coherent interactive visualization. *Proc. CHI* (2004), 105–112.

5. Battista, G., Algorithms for drawing graphs: an annotated bibliography. *Computational Geometry 4*, 5 (1994), 235–282.

6. Bostock, M., and Heer, J., Protovis: a graphical toolkit for visualization. *IEEE Trans. Vis. Comput. Graphics 15*, 6 (2009), 1121–8.

7. Burch, M., Beck, F., and Diehl, S., Timeline trees: visualizing sequences of transactions in information hierarchies. *Proc. AVI* (2008), 75–82.

8. Carlis, J. V., and Konstan, J. A., Interactive visualization of serial periodic data. *Proc. ACM UIST* (1998), 29–38.

9. Chittaro, L., Visualization of patient data at different temporal granularities on mobile devices. *Proc. AVI* (2006), 484.

10. Cousins, S., and Kahn, M., The Visual Display of Temporal Information. *Artificial intelligence in medicine 3*, 6 (1991), 341–357.

11. Fails, J., Karlson, A., Shahamat, L., and Shneiderman, B., A Visual Interface for Multivariate Temporal Data: Finding Patterns of Events across Multiple Histories. *Proc. IEEE VAST* (2006), 167–174.

12. Fekete, J.-D., The InfoVis Toolkit. *Proc. IEEE Symp. Information Visualization* (2004), 167–174.

13. Guerra Gómez, J., Wongsuphasawat, K., Wang, T. D., Pack, M. and Plaisant, C., Analyzing Incident Management Event Sequences with Interactive Visualization. Transportation Research Board 90th Annual Meeting Compendium of Papers (2011).

14. Harrison, B., Owen, R., and Baecker, R., Timelines: an interactive system for the collection and visualization of temporal data. *Graphics Interface* (1994), 141.

15. Havre, S., Hetzler, E., Whitney, P., and Nowell, L., ThemeRiver: visualizing thematic changes in large document collections. *IEEE Trans. Vis. Comput. Graphics 8*, 1 (2002), 9–20.

16. Hewagamage, K., Hirakawa, M., and Ichikawa, T., Interactive visualization of spatiotemporal patterns using spirals on a geographical map. *Proc. IEEE Symp. Visual Languages* (1999), 296–303.

17. Johnson, B., and Shneiderman, B., Tree-maps: A space-filling approach to the visualization of hierarchical information structures. *Proc. IEEE Vis* (1991), 284–291.

18. Karam, G., Visualization using timelines. *Proc. ACM SIGSOFT Int'l Symp. Software testing and analysis* (1994), 125–137.

19. Klimov, D., Shahar, Y., and Taieb-Maimon, M., Intelligent visualization and exploration of time-oriented data of multiple patients. *Artificial intelligence in medicine 49*, 1 (2010), 11–31.

20. Kruskal, J., and Landwehr, J., Icicle Plots: Better Displays for Hierarchical Clustering. *The American Statistician 37*, 2 (1983), 162–168.

21. Lamping, J., The Hyperbolic Browser: A Focus+Context Technique for Visualizing Large Hierarchies. *Journal of Visual Languages & Computing 7*, 1 (1996), 33–55.

22. Letunic, I., and Bork, P., Interactive Tree Of Life (iTOL): an online tool for phylogenetic tree display and annotation. *Bioinformatics 23*, 1 (2007), 127–8.

23. Lin, J., Keogh, E., and Lonardi, S., Visualizing and discovering non-trivial patterns in large time series databases. *Information Visualization 4*, 2 (2005), 61–82.

24. Munzner, T., H3: laying out large directed graphs in 3D hyperbolic space. *Proc. VIZ '97* (1997), 2–10.

25. Phan, D., Paepcke, A., and Winograd, T., Progressive multiples for communication-minded visualization. *Proc. Graphics Interface (GI)* (2007), 225.

26. Plaisant, C., Mushlin, R., Snyder, A., Li, J., Heller, D., and Shneiderman, B., LifeLines: using visualization to enhance navigation and analysis of patient records. *Proc. AMIA Symp.* (1998), American Medical Informatics Association, 76.

27. Reingold, E., and Tilford, J., Tidier Drawings of Trees. *IEEE Trans. Softw. Eng.*, 2 (1981), 223–228.

28. Robertson, G. G., Card, S. K., and Mackinlay, J. D., Information visualization using 3D interactive animation. *Comm. of the ACM 36*, 4 (1993), 57–71.

29. Shneiderman, B., The eyes have it: a task by data type taxonomy for information visualizations. *Proc. IEEE Symp. Visual Languages* (1996), 336–343.

30. Shneiderman, B., Extreme visualization: squeezing a billion records into a million pixels. *Proc. ACM SIGMOD Conf. Management of data* (2008), 3–12.

31. Shneiderman, B., and Plaisant, C., Strategies for evaluating information visualization tools. *Proc. AVI BELIV* (2006), 1.

32. Stasko, J., and Zhang, E., Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. *Proc. IEEE Symp. Information Visualization* (2000), 57–65.

33. Vrotsou, K., Ellegard, K., and Cooper, M., Everyday Life Discoveries: Mining and Visualizing Activity Patterns in Social Science Diary Data. *Proc. Int'l Conf. Information Visualization* (2007), 130–138.

34. Vrotsou, K., Johansson, J., and Cooper, M., ActiviTree: interactive visual exploration of sequences in event-based data using graph similarity. *IEEE Trans. Vis. Comput. Graphics 15*, 6 (2009), 945–52.

35. Wang, T. D., Plaisant, C., Quinn, A. J., Stanchak, R., Murphy, S., and Shneiderman, B., Aligning temporal data by sentinel events: discovering patterns in electronic health records. *Proc. CHI* (2008), 457–466.

36. Wang, T. D., Plaisant, C., Shneiderman, B., Spring, N., Roseman, D., Marchand, G., Mukherjee, V., and Smith, M., Temporal Summaries: Supporting Temporal Categorical Searching, Aggregation and Comparison. *IEEE Trans. Vis. Comput. Graphics 15*, 6 (2009), 1049–1056.

37. Weber, M., Alexa, M., and Muller, W., Visualizing time-series on spirals. *Proc. IEEE Symp. Information Visualization* (2001), 7–13.

38. Wetherell, C., and Shannon, A., Tidy Drawings of Trees. *IEEE Trans. Softw. Eng.*, 5 (1979), 514–520.

39. Wongsuphasawat, K., and Shneiderman, B., Finding comparable temporal categorical records: A similarity measure with an interactive visualization. *Proc. IEEE VAST* (2009), 27–34.