J Med Syst (2011) 35:1135–1152
DOI 10.1007/s10916-011-9718-x

ORIGINAL PAPER

Extracting Insights from Electronic Health Records: Case Studies, a Visual Analytics Process Model, and Design Recommendations

Taowei David Wang · Krist Wongsuphasawat · Catherine Plaisant · Ben Shneiderman

Received: 17 January 2011 / Accepted: 13 April 2011 / Published online: 4 May 2011

Abstract Current electronic health record (EHR) systems facilitate the storage, retrieval, persistence, and sharing of patient data. However, the way physicians interact with EHRs has not changed much. More specifically, support for temporal analysis of a large number of EHRs has been lacking. A number of information visualization techniques have been proposed to alleviate this problem. Unfortunately, due to their limited application to a single case study, the results are often difficult to generalize across medical scenarios. We present the usage data of Lifelines2 (Wang et al. 2008), our information visualization system, and user comments, both collected over eight different medical case studies. We generalize our experience into a visual analytics process model for multiple EHRs. Based on our analysis, we make seven design recommendations to information visualization tools to explore EHR systems.

Keywords Information visualization · Medical records [L01.280.900.968] · User-computer interface [L01.224.900.910] · Computer-assisted decision making [L01.700.508.100] · Computer graphics [L01.224.108]

Introduction

Whether it is to diagnose a single patient or to obtain quality assurance measures of health care by analyzing multiple patients, physicians and clinical researchers must incorporate large amounts of multivariate historic data. Current electronic health record (EHR) systems facilitate the storage, retrieval, persistence, and sharing of patient health information; however, the availability of information does not directly translate to adequate support for complex tasks physicians and clinical researchers encounter every day.

Overwhelmingly large amounts of information and a lack of support for temporal queries and analyses are but a few problems physicians and clinical researchers face. As a result, a number of information visualization systems have been introduced to address these issues. These systems support higher-level decision-making and exploratory analysis tasks in the medical domain. Commendably, these systems aim to solve real problems physicians face and to add value to the EHR systems for the end-users. However, these systems are often designed for one specific medical scenario, and subsequently evaluated on that scenario. As a result, it is difficult to make generalizations on physicians' visual analytics process or the process's user requirements in EHRs.

In contrast, our information visualization tool, *Lifelines2*, has been applied to 11 different case studies, eight of which are in the medical domain. By case studies, we mean a long-term, in-depth study on our users' usage and experience of Lifelines2 on a domain and data sets that they select and care about. Case studies provide human-computer interaction (HCI) researchers a valuable perspective on how their tools are used in the real world, as opposed to in an experimental setting. This differs from medical case studies. Since Lifelines2's inception, we have worked closely with physicians and hospital administrators to gather user requirements for the tasks of temporal search and exploratory analysis of multiple patient records over time. It has been used by physicians for the purpose of (1)

T. D. Wang · K. Wongsuphasawat · C. Plaisant · B. Shneiderman
Department of Computer Science & Human-Computer Interaction Lab, University of Maryland,
College Park, MD 20742, USA

T. D. Wang (✉)
1 Constitution Center OCC-02-260L
Boston, MA 02129, USA
e-mail: taowei@gmail.com

obtaining quality assurance measures, (2) assessing impact on patient care due to hospital protocol changes, (3) replicating published clinical studies using in-hospital data, and (4) simply searching for patients with interesting medical event patterns.

Over the two-and-half year period in which these case studies took place, we observed how physicians used Lifelines2, logged the actions performed and features used, and collected physicians' comments. By analyzing the observations, logs, and user-feedback, we were able to make generalizations about searching for temporal information in EHRs. Related work first presents related work. Lifelines2 introduces Lifelines2 and describes one case study in detail. We then present an analysis of Lifelines2 log data and a process model, and conclude with a list of design recommendations. Further information and video demonstrations are available at http://www.cs.umd.edu/hcil/lifelines2.

Related work

As EHR systems become more prevalent, the need for effective techniques to interact with EHRs also becomes more pressing. A growing number of recent field research efforts have studied how end-users interact with EHRs in hospitals. While some studies have focused on how patients can benefit from a display of their own EHR [1], most efforts have focused on how medical professionals behave as end users. These studies follow, for example, physicians' workflow in supplementing, annotating, and reusing EHRs [2–5]. These field studies identify important design challenges, which EHR systems designers must overcome to support medical professionals' tasks. Unfortunately, the field studies often fall short of recommending possible technologies for solving these problems [2, 4, 5].

Many EHR systems lack features that support important end-user tasks. Exploratory analysis, effective representation, and temporal queries are but a few that are often found lacking even in state-of-the-art systems such as Amalga [6] or i2b2 [7]. As a result, many information visualization systems have been proposed with different techniques to support these tasks and supplement the EHR systems. These systems focus on novel ways to integrate and visualize personal information in a useful way. The integration aspect is akin to creating personal histories or life narratives [8–10] to better contextualize medical information. The visualization techniques vary widely for different use cases. Some approaches are static visualizations, such as the one proposed by Powsner and Tufte [11], but most modern ones are interactive. Many of these support only a single EHR—Lifelines [12], Midgaard [13], Web-Based Interactive Visualization System [14], VIE-VISU [15], to name a few. They generally focus on supporting physicians to quickly absorb a patient's potentially lengthy medical history in order to make better medical decisions. On the other hand, a number of systems expand the coverage to multiple EHRs, for example, Similan [16], Protempa [17], Gravi++ [18], VISITORS [19], and IPBC [20]. These systems typically focus on novel search and aggregation strategies for multiple EHRs.

These information visualization systems are all motivated by real issues physicians or clinical researchers encounter when the typical presentation of medical data is not conducive to their analysis tasks. However, because of limited availability of physicians and clinical researchers, very few systems have gone through multiple detailed long-term case studies [21]. While these systems demonstrate the usefulness of their features in one or two isolated medical case studies, the results are harder to generalize. As a consequence, these information visualization efforts rarely make broader generalizations about their techniques. They also rarely make recommendations on the directions information visualization designers for EHRs should pursue further. In contrast, we applied Lifelines2 to 11 case studies, eight of which are medical scenarios according to the multidimensional in-depth long-term case studies (MILC) model [22]. By analyzing the multidimensional user and usage data, we believe we can contribute to the field by making useful generalizations and recommendations. However, because Lifelines2 aims to support searching and exploring multiple EHRs, the generalizations and recommendations presented in this work may not apply to the design of single-EHR systems.

In addition to presenting the analysis of user and usage data of Lifelines2, we also present a process model which generalizes how physicians seek information in EHRs. Our process model is similar in construction to the sense-making loop presented by Stuart Card and others [23–25]. However, ours differ in the level of granularity and application domain. We focus specifically on multiple EHRs and with a strong emphasis on temporal analysis. Our level of granularity and task-specificity is similar to the proposed process model for social network analysis [26].

Lifelines2

Lifelines2 is designed for visualizing temporal categorical data for multiple records. Temporal categorical data are time-stamped data points that are not numerical in nature. For example, in an EHR, the patient's past hospital visits, diagnoses, treatments, medication prescribed, medical tests performed, etc. can be considered as temporal categorical data. These data are point data (no durations) with a name, and can be thought of as "events." This differs from temporal numerical data such as blood pressure readings, or platelet counts. Lifelines2 visualizes these temporal categorical data and provides a number of visualization and interaction techniques for exploratory analysis.

Figure 1 shows a screen shot of Lifelines2, in which region (a) is Lifelines2's main display of EHRs. Each patient occupies a row, and is identified by its ID on the left. Under the ID, a list of event types in that EHR is listed. Each event is represented by a color-coded triangle and placed on the time line. Region (c) is the control panel for Lifelines2. Each patient is *Aligned* by the 1st occurrence of *IMC*, *Ranked* by the number of *ICU* events, and *Filtered* by the sequence of events [*Admit*, *No ICU*, *IMC*, *ICU*]. EHRs that match the *Filter* are highlighted in orange. Of the 318 EHRs in Fig. 1, only 39 were found to be matches. Region (b) is called a *temporal summary*, and it displays the distribution of *Admit*, *Exit*, and *ICU* events. In (a) and (b), analysts can zoom in, zoom out, pan, and scroll. Tool tips provide detailed information for each event when moused over.

By *Aligning* every patient by its corresponding events (1st, 2nd, …, and last, 2nd to the last, …), physicians can better compare the patients as a group. Events that occur commonly before or after the *Alignment* can be more easily detected. When an *Alignment* is active, the time line becomes relative to the *Alignment* (Fig. 2b). Analysts can also *Align* by *all* occurrences of an event type, in which case Lifelines2 duplicates each EHR by the number of events of that type the EHR contains, and shifts the duplicates by each of the event instances.

Finally, analysts can *Rank* the EHRs by their ID (default behavior), or by the number of occurrences of the different event types, such as the number of ICU visits. They can also *Filter* by the number of occurrences of event types, or by a sequence *Filter* (Fig. 1c). *Align*, *Rank*, *Filter* are affectionately called the ARF framework, and serves as a basis for user interaction in Lifelines2 [27].

Temporal summaries [28] are histograms of events over time. Additionally, analysts can change number of events to number of EHRs or number of events per EHR over time. Temporal summaries are temporally synchronized with the main visualization, and share the same temporal granularity. Analysts can use direct manipulation to select EHRs that contribute to a certain bin in the histogram. By combining

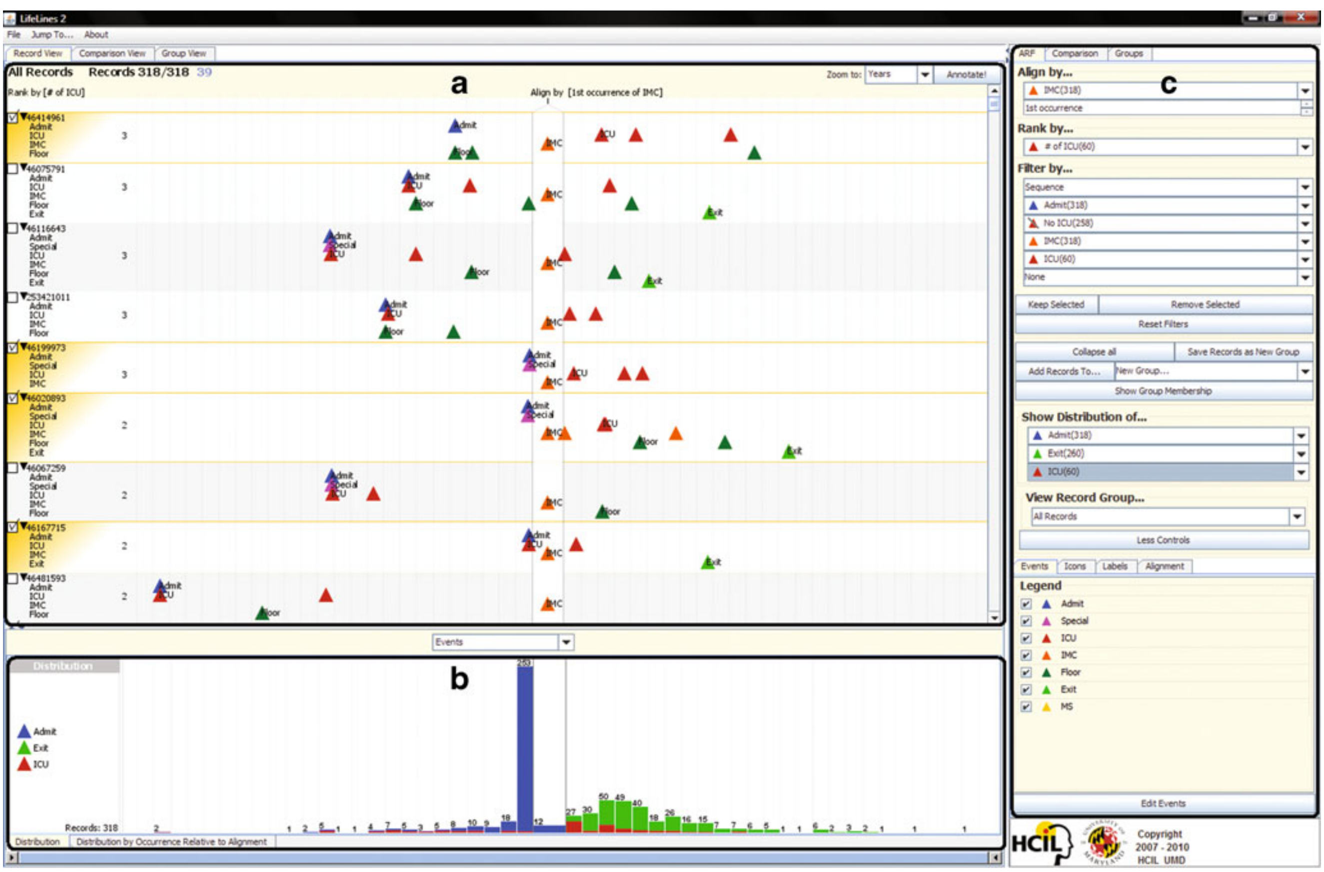
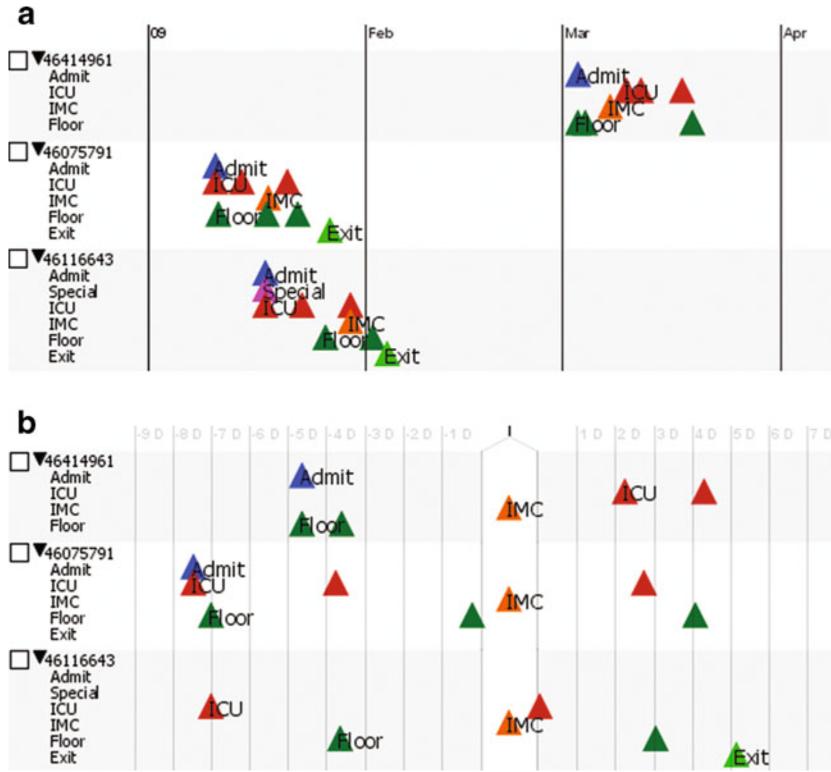


Fig. 1 A screen shot of Lifelines2. **a** shows the main visualization of multiple EHRs. **b** is a temporal summary, showing the distribution of the three event types *Admit*, *Exit*, and *ICU* over time. **c** is the control panel for Lifelines2. Each of the 318 patients is *Aligned* by their 1st occurrence of *IMC*, *Ranked* by the number of *ICU* events, and *Filtered* by the sequence of events

Fig. 2 Part (**a**) shows three EHRs in Lifelines2 that are un-aligned (calendar time). Part (**b**) shows the same three EHRs aligned by their 1st *IMC* event (relative time)



alignment and temporal summaries, analysts can select, for example, all patients that entered the *ICU* within 24 h of entering the *IMC*.

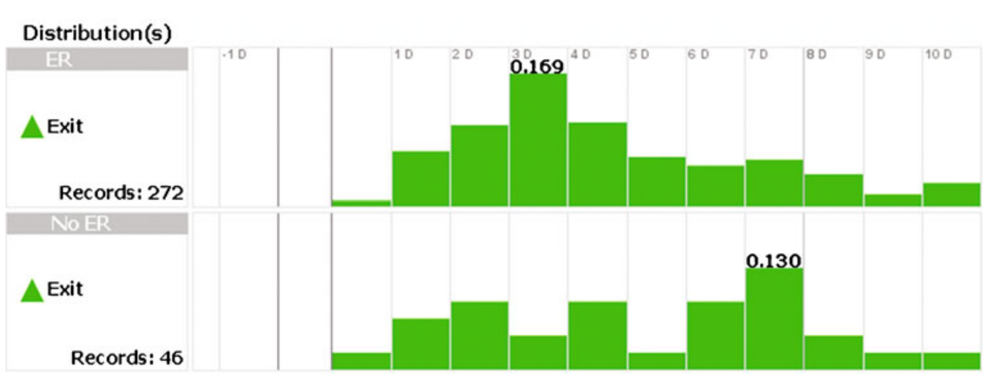
After *Filtering* and selection, analysts can optionally save their results as a separate group. Set operations are available to create the union, intersection, and difference on the groups. Multiple groups can be compared in comparison mode, where one temporal summary represents a group, and arbitrarily many groups can be compared. Figure 3 compares two groups of patients by their hospital exit events over time. The first contains patients who have entered the emergency room (ER), and the second contains those who have not. These patients are all *Aligned* by their admission time (*Admit*), and the distribution of *Exit* events is plotted. The events are normalized by the number of patients in each group, subsequently the bars represent the percentage of patients who exit in each day following their admission. There is a peak for patients who go through the ER, while those who do not have a more irregular distribution. The comparison features allow physicians to, for example, directly compare patient groups that undergo different treatment options.

Medical case studies

Overview

We conducted our case studies in two phases. In the first phase (early-adoption), physicians and hospital administrators worked with us to iteratively refine and improve Lifelines2's features and usability. Our collaborators

Fig. 3 Comparison of the distribution of the *Exit* events for two different groups of patients



learned features of Lifelines2. This early adoption phase lasted for over a year, during which we conducted three case studies: (1) finding patients who exhibited contrast-induced nephropathy, (2) finding patients who exhibited heparin-induced thrombocytopenia, and (3) studying hematocrit levels in trauma patients with respect to length of stay in the hospital and discharge patterns.

After the early-adoption case studies, we conducted eight additional mature-adoption case studies, five of which were in the medical domain. In this phase, no new novel interaction or visualization features were implemented in Lifelines2. We only added bug fixes and small features that facilitate the analyses. In this phase, Lifelines2 was used for a number of different analysis tasks: (1) Replicating a study [29] that investigates the relationship between day light savings time change and heart attack incidents using clinical data. (2) Performing a follow-up study on heparin-induced thrombocytopenia in ICU patients [28], (3) studying hospital room transfer patterns as a measure for quality assurance (two case studies, one for the *Bounce-Back* patterns, and the other for the *Step-Up* patterns), and (4) studying the impact on patient care due to a change in protocol that governs when Bi-level Positive Airway Pressure (BiPAP) is applied. Table 1 shows the statistics of selected datasets. While sequences of temporal events often temptingly suggest causality, our case studies are all based on some well-described phenomena (contrast-induced nephropathy, heparin-induced thrombocytopenia, or the step-up patterns, etc.). No causality is otherwise suggested. Instead we emphasize that Lifelines2 is a tool designed for discovery and search of temporal event patterns. With that said, we believe discovery is an important step that may eventually lead to establishing causal links with well-designed controlled experiments.

All of our case studies were selected and initiated by our collaborators (regardless of phase). Our collaborators would present a medically relevant question that they would like to investigate. These questions are typically difficult to answer with their current EHR system and supporting software. However, not all questions are good candidates. For example, questions that involve analysis of numerical data, for which Lifelines2 is ill-suited, are discontinued (such as the hematocrit study). Most of the unsuitable questions arise during the first phase, when collaborators' familiarity with Lifelines2 is low. By the end of the early-adopter phase, however, our collaborators became experts with Lifelines2's features, and, subsequently, became very good at identifying interesting medical questions suitable for Lifelines2. Due to time constraints, however, we were only able to perform five mature-adoption case studies.

Each case study follows the same template. Physicians describe a medical scenario interesting to them and ask database administrators to obtain the relevant data from their current EHR system. The data is preprocessed and then converted to Lifelines2 format. The data is later loaded in Lifelines2 and interactively explored together by our collaborators and us (University of Maryland (UMD) researchers). This exploration often revealed additional problems, which may, for example, require additional data or prompt additional preprocessing. It usually takes two to three one-hour meetings with our collaborators to ensure good data quality, followed by more meetings dedicated to analysis.

During the analysis meetings, the physicians and UMD researchers share a large display. In the early-adoption phase, we encourage physicians to interact with Lifelines2 directly to (1) familiarize themselves with the features and operations of the system, and (2) identify bugs and interface issues as end-users. In the mature adoption phase, the physicians would typically dictate what actions to take, and UMD researchers would interact with Lifelines2 based on the dictation. Using this methodology, we were able to better follow our collaborators' thought process in a field we were unfamiliar with. This also forced our collaborators to explain to us the medical significance and nuances of their interpretation. During these meetings, we recorded our collaborators' feedback. The feedback typically included our collaborators' impressions of Lifelines2, its comparison and contrast with their current EHR system, and suggestions of features to include in future versions. They also often include discussions of the case study and proposals of additional related case studies. The recording was originally collected via note-taking and later via audio recording. All interactions performed in Lifelines2—*Align*, *Rank*, *Filter*,

Table 1 Basic statistics on the selected datasets. The numbers of event patterns and the average length of such patterns in each case study are also shown

Dataset name	#Records	#Events	# Patterns	Average length of patterns
Creatinine	3598	32134	5	3.2
Heparin	841	65728	5	4.6
Heart attack	9361	196581	–	–
Transfer	51006	207187	5	3
BiPAP	6583	135951	9	3.89
Step-up	284[a]	1612[a]	1	3
Bounce-back	544[a]	3055[a]	10	3

[a] average number over many quarterly datasets

Zoom, etc.—are logged automatically using Lifelines2's logging facility.

In the early stages of Lifelines2 we worked with a neurologist, an osteopathic physician, and two nursing professors. Over the eight medical case studies, the only medical professionals we worked with were physicians, including an emergency room director, two professors of medicine, an internal medicine physician, and a resident. Some of them participated in more than one case study. Database administrators and EHR system engineers were also involved. A case study typically takes one to 6 months to complete. Some case studies include repeated analysis of patients in different time periods, while some are performed for a single period. Some case studies include patients during a 10 year period, while others include only a few months. The number of patients in the case studies can be as few as a couple hundred (with a few thousand events) or as many as 51,000 (with over 207,000 events), and the event-per-record ratio is different for each case study. Case studies can take tens of meetings and hundreds of e-mail exchanges to organize, execute, and finally compile final results.

Case study: Identifying step-up patterns

We present a case study on patient room transfers in detail to demonstrate how Lifelines2 is used in a real scenario. Hospital rooms can be roughly classified into: (1) *ICU* (intensive care units that provide the highest level of care), (2) *IMC* (intermediate medical care rooms that house patients who need elevated level of care, but not serious enough to be in *ICU*), (3) *Floor* (normal hospital beds that typically house patients with no life-threatening conditions), and (4) *Special* (emergency room, operating room, or other rooms). In this study, the dataset also includes patients' hospital admission (*Admit*) and hospital discharge (*Exit*) if they have already exited. Each of these room event data comes with a time stamp, indicating when the patient is transferred-in. Transfer-out is implied by subsequent transfer-ins to another rooms or *Exit*.

The physicians are interested in the *Step-Up* pattern. This is a pattern where a patient initially triaged to go to an IMC room escalates to ICU rooms immediately. The pattern may be indicative of mis-triage—that is, sending patients too sick to IMC instead of ICU in the first place. The exact criteria are patients who were sent to IMC and escalated to ICU within 24 h. For example, the fifth patient from the top in Fig. 1 exhibits exactly the Step-Up pattern.

There were two hypotheses our physician collaborators are interested in. First, the nurses in IMC had noticed anecdotally that Step-Ups have increased. Our collaborators wanted to verify this claim and decide if protocols for performing triage need to be changed. Secondly, our collaborators hypothesized that because newly graduated doctors enter the hospital in the third quarter (July-September) every year, the percentage of Step-Up cases might be higher in these months due to their inexperience.

The original query seemed easy to perform at first. By first *Aligning* by all patients' *IMC* events and selecting all *ICU* events that occur within 24 h after the *Alignment*, we should be able to identify all patients who exhibit the Step-Up case. However, when the authors and our collaborators examined data together, we realized several issues. For example, the pattern must not include any *Floor* events between *IMC* and *ICU* (transferring from *IMC* to *Floor* then to *ICU*) because this suggests the escalation from *Floor* to *ICU* is likely not due to an earlier triage. Similarly, there should not be an *ICU* prior to the *IMC* in question. If there were, the patient was already in *ICU*, and this would not be considered a Step-Up. These nuances in data were not expected initially, but the visualization and the application of *Alignment* made their existence alarmingly obvious; whereas a direct application of, for example, SQL would have made the discovery and the correction difficult.

We first presented the Step-Up case study in an earlier publication [30]. However, through continued verification to our analysis process, we discovered a few mistakes in our conversion file and new information from our collaborators. For example, our collaborators told us that there is the addition of the *MS* room, an overflowing area for ICUs. This means that when we account for transfers into *ICU* in the Step-Up pattern, we must include patients going to *ICU* or *MS*. We modified our conversion script, reconverted the raw data into Lifelines2 format, and re-performed the analysis. While the main results have not changed, the specific numbers have. We then apply the following interactions in Lifelines2 to identify the Step-Up cases:

1. Perform a sequence *Filter* using [*IMC*, *No Floor*, *ICU*}], and save the results as a new group named *IMC-No Floor-ICU*.
2. Perform a sequence *Filter* using [*IMC*, *No Floor*, *MS*}], and save the results as a new group named *IMC-No Floor-MS*.
3. Use the *Union* operation in Lifelines2 to combine the two groups into a few one: *Potential Step-Ups*.
4. *Align* by the *1*st occurrence of *IMC*.
5. Temporally select (in a temporal summary) *ICU* events that occur any time prior to the *Alignment*, and remove the selected EHRs.
6. Temporally select *ICU* and *MS* events that occur within 24 h after *Alignment*, and keep the selected EHRs.
7. Save as a new group and export this new group as a file.
8. Return to group *Potential Step-Ups*.
9. Repeat steps 4–8 by changing the *1*st *IMC* to the n^{th} *IMC*. Stop when there are no records with *n* *IMC*s.

We conducted this study for every quarter from January, 2007 to March, 2010. Each quarter took roughly 12–20 min to perform. The data contains all patients who have been admitted to *IMC* in that period. A screen shot of a quarterly data is shown in Fig. 1. Figure 4 shows the number of patients admitted to IMC in that period. The IMC patient count is overall on a rising trend because of hospital expansion. Figure 5 shows the number of patients who exhibit the Step-Up pattern in the same period, a subset of all patients admitted to IMC. The graph looks more jagged than Fig. 4, but there does also seem to have a upward trend. Finally, we plot the percentage of patients who exhibit Step-Up patterns (out of the IMC patient counts) in Fig. 6. The percentage of Step-Up patients peaks in the second quarter of 2008 (at nearly 8%), but has mostly been holding steady between 4 and 7%. One of our physician collaborators explains, “The nurses must have gotten the impression that mis-triaging occurred more often because they have encountered more Step-Up cases. They felt the increased number of cases was due to errors in the triaging, while the real reason is more likely due to the increase of IMC patients.” He also added, “The reason for the increase of IMC patients was not due to the increase of diseases or injuries. Instead, it was merely a reflection on the expansion of IMC care in the hospital.”

The average percentage of Step-Up cases for each quarter is shown in Fig. 7 to investigate the second hypothesis. Of the four quarters, the second quarter has the highest percentage of Step-Up cases (6.55%), and the first quarter has the fewest (5.42%). There is no evidence of an increase of Step-Up cases in quarter 3. One physician collaborator commented that, “The attending physicians (supervisors of the residents) must have been doing a good job reviewing the results of the resident triaging process.” He, however, did not offer an explanation for why the numbers in the first quarter are so much lower than the others or that the second quarter has much higher percentage.

Our collaborators have taken the results produced from our collaboration using Lifelines2 (patient counts in Excel spread sheets, graphs, Lifelines2 screen shots, and annotation) to a physicians’ meetings in the hospital. The consensus is that these percentages are well-within the boundaries, and changing triage procedures is probably not a necessary course of action at this point. They note that the analysis is interesting, and would love to keep performing the same analysis for as long as possible to build an historical baseline and so every quarter in the future can be evaluated the same way. Finally, as one collaborators notes, the data can be used to compare to the numbers from other hospital care systems, for the purpose of hospital metrics.

User experience and feedback

During the early-adoption case studies, several physicians gave good appraisal to the visual representation in Lifelines2. One collaborator said, “I am a very visual person. To be able to see the patient records this way allows me to understand it so much quicker and more reliably.” Later when interviewed by *Terp Magazine*, he commented, “This technology saves time and gives us another important diagnostic tool”, and “[it] will not only make for better care by doctors, but also help patients make healthier choices on their own” [31]. Another collaborator chimed in about the *Alignment* feature, “This is great. To be able to see what occurs before and after a heart attack in all these patients is great.”

While during the mature adoption case studies, our physician collaborators dictated and did not directly interact with Lifelines2 in the majority of the collaborative analysis sessions, they did sometimes drive the application and gave feedback on using Lifelines2. In the Step-Up case, for example, one collaborator drove the application alone a few times. Through a hospital initiative to explore novel technologies that can help improve patient care, he was able to work closely with the UMD researchers far more frequently than other physicians.

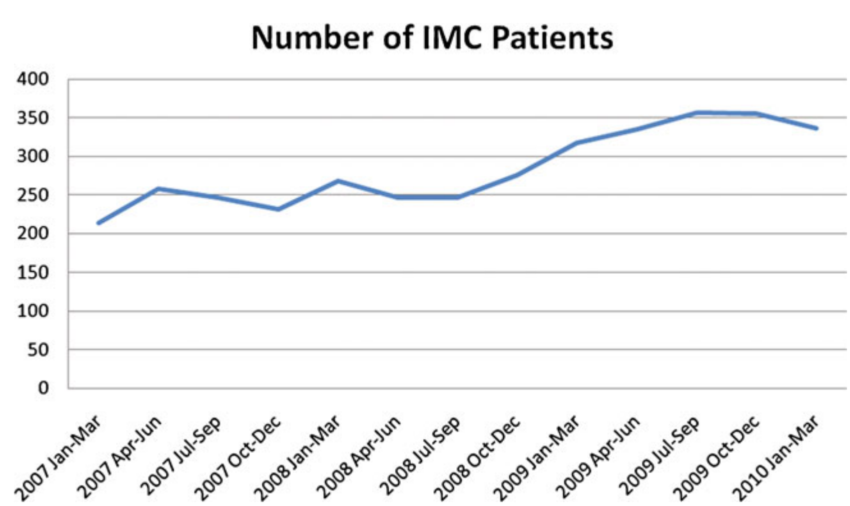


Fig. 4 The quarterly number of patients who entered IMC for the period of January 2007 to March 2010

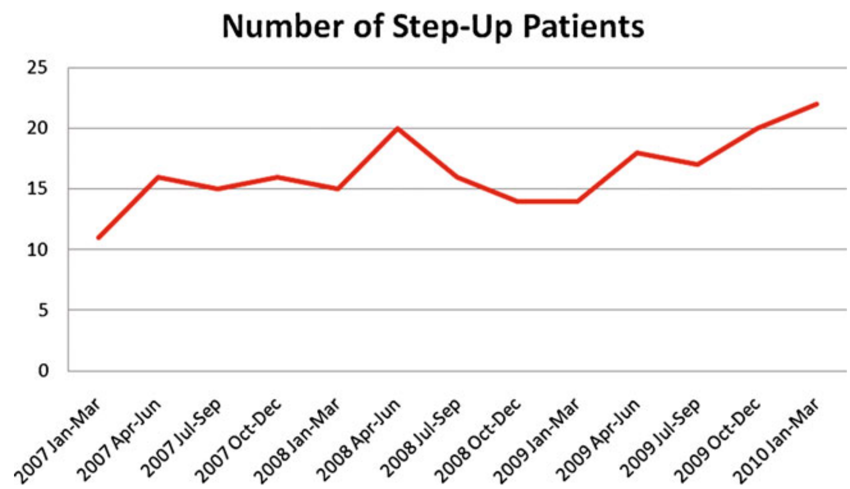


Fig. 5 The quarterly number of patients who exhibited the Step-up pattern for the period of January 2007 to March 2010

By the time we started conducting the Step-Up case study, we had already met a few times, and he became fairly familiar with Lifelines2. We observed that he had no problem using *Align*, *Group Selection/Creation*, *Rank*, *Temporal Summaries*, and *Selections on Temporal Summaries*. However, he did have a problem formulating his query into a sequence *Filter*. He mentioned that he did not see the sequence *Filter* control immediately on screen so it did not remind him how to get to the *Filter*. After we showed him how to get to the sequence *Filter*, he was able to perform the necessary queries for Step-Up in a matter of minutes by himself.

Over all, he felt that "Lifelines2 would save me so much time to deal with all the different scenarios.", "I would never have to spend hours to write broken Excel scripts that produce low-quality data ever again!". He also commented that, "the good thing about Lifelines2 is its visual power […] I can visually look quickly to see if there is anything amiss," and "I can perform the pattern-finding in matter of minutes, and feel that data is far more reliable [than the Excel spreadsheet he had created] at the same time." Finally, having spent several months working with Lifelines2 also changed how he views clinical data. In particular, when I asked him about Lifelines2, he said that, "Yeah, alignment was very different idea, yet so natural! I think about clinical problems in terms of alignment now, but none of my coworkers does the same."

Throughout the interaction with domain experts, we observed that temporal ordering and aggregation of events often elicit domain knowledge, resulting in a fuller description of a patient's situation than the data would convey to a non-domain expert. For example, seeing a patient with annual attacks of asthma and pneumonia every autumn led a nursing professor to surmise that the patient may be an elderly or someone vulnerable to seasonal flu [27]. In the contrast-induced nephropathy case, the emergency room director could identify patients who probably had chronic kidney problems [28]. The extent and frequency of contextual domain knowledge being recalled is difficult to quantify. However, it is widely believed in the information visualization field that the appropriate representation of data can lead to better interpretation [12, 13, 21], as these two examples illustrate how a physician may apply domain knowledge to suggest patients' situation not included in plain data.

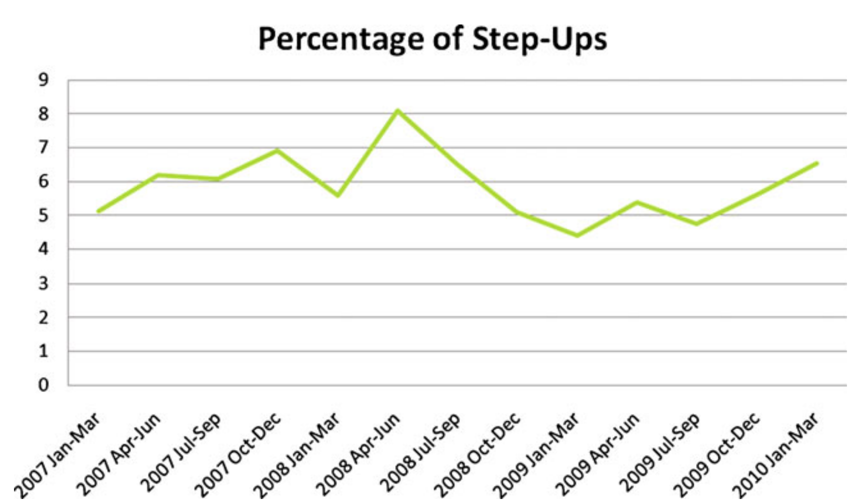


Fig. 6 The percentage of the Step-Up patients over all IMC patients in the period of January 2007 to March 2010

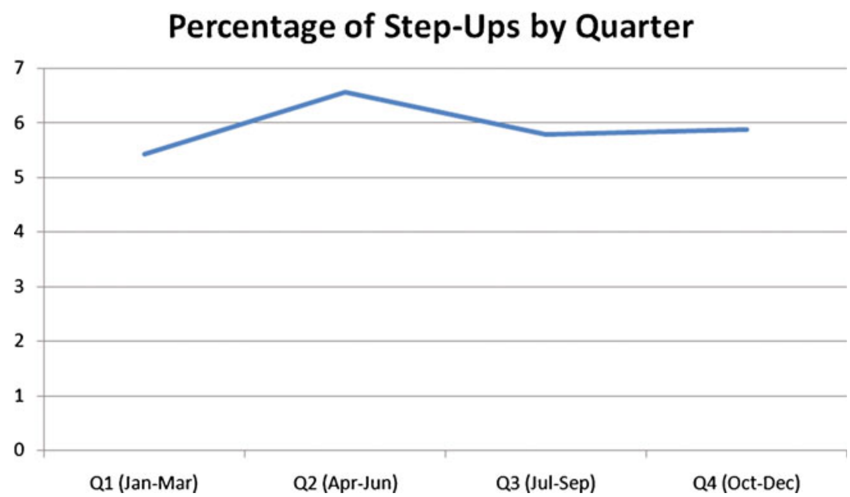


Fig. 7 The average percentage of Step-Up cases over all IMC patients in the period of January 2007 to March 2010

Despite all the positive feedback, several user interface issues were identified in the mature adoption case studies. First, the *align-by-all* operator can be conceptually confusing. While analysts are presented with *instances* (duplicates of records), Lifelines2 operates on records. This inconsistency is a major design problem because the visual representation differs from what is happening in the data. Since the first meetings on the Bounce-Back and Step-Up studies, this critical issue has been fixed. Secondly, some case studies such as the Bounce-Back study and the Step-Up study require the merging of a number of groups. Initially we perform merging outside of Lifelines2, but that soon becomes a burden. New functionalities are then implemented to support typical set operations: union, intersection, and difference to facilitate analysis. Finally, features that support manually creating, adding, or removing records to/from a group are also implemented to support manual review of the records in case studies such as BiPAP. Other smaller features such as search-by patient ID, data export, group import, annotation, and screen capture are added to facilitate the overall analysis process. Some features our collaborators would like are left out. For example, the dream feature for our collaborator on the Step-Up case study is automation of an analysis, as he intends to perform the same study for every quarter longitudinally. He wants it to run the analysis as a script, and at the end allow him the freedom to visually inspect the results so that he can manually change the steps of analysis if he needs to. For example, if his review of the process revealed something wrong, he would like to allow for different branches of the analysis to take place.

Implementation details and system performance

Lifelines2 is developed entirely in Java. It contains over 27,000 lines of code, and uses the Piccolo 2D graphics library [32]. All performance measures in this section are obtained from running Lifelines2 on a 2.4 GHz Intel Core2 Duo laptop running Windows Vista Home Premium Edition 64-bit, with 4 GB of RAM. As an inspirational prototype, Lifelines2 is not tightly coupled with an underlying database system, which allows us to better work with collaborators who may use a variety of different storage technologies. Lifelines2 takes in a simple three-column text file, where the first column lists the patient IDs; the second lists the event type; and the third lists the time stamps. Each row thus describes one event that belongs to a patient. This format is designed to be easily created by any database system or spreadsheet. Because Lifelines2 is not connected to a database, the amount of data it can load is limited to amount of RAM available on the machine (and assigned to the Java Virtual Machine). In our tests, Lifelines2 is able to load up to over 100,000 records with over 1,500,000 events when the Java Virtual Machine is allocated 1 GB of RAM.

The main functionalities in Lifelines2 are its visualization and its interaction techniques. In order to provide a sense of fluidity when using the system, much work has been dedicated to make sure the drawing and the interaction are optimized. For example, drawing of the EHRs only occurs for visible EHRs and developing a novel pattern search algorithm [33]. The drawing speed is determined by the number of events that appear on screen, while the query (performing *Align-Rank-Filter*) speed is determined by the number of records in the set, and the density of events. Figure 8 shows a comparison of Lifelines2's query time and render time over five dataset of different sizes. The data points represent the average of ten different trials. A linear fitting is performed to show the slope. Query time outpaces render time quickly as the data size grows. Analyzing the details of the trials reveals also that the query time has twice as much standard deviation as the render time. The bound at which a user experiences a noticeable delay is if an action or redraw takes longer than 160 milliseconds. This is only achievable for data size of around 10,000 records for querying

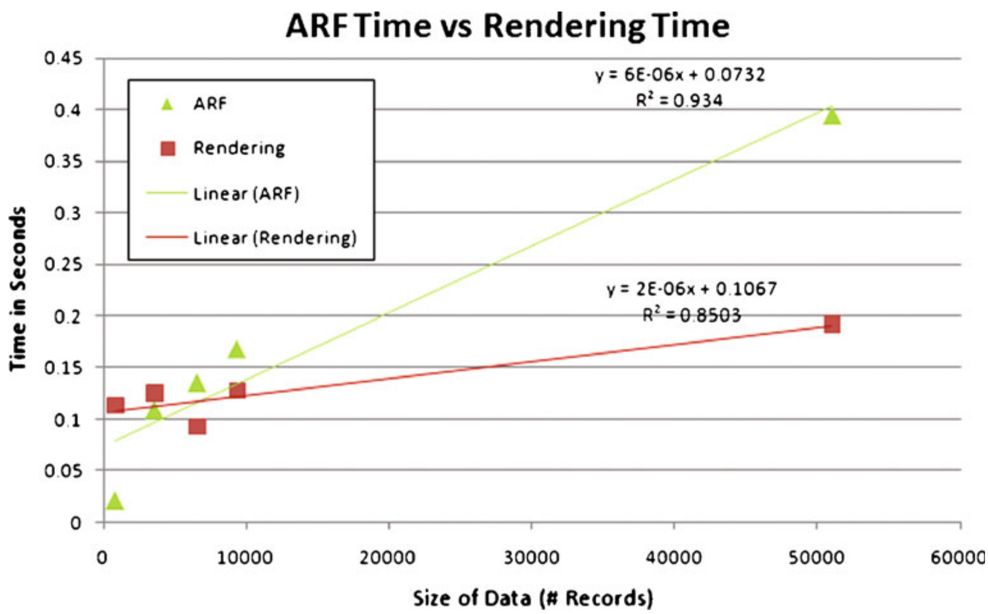


Fig. 8 The time growth comparison for ARF (query) time, and render time in Lifelines2 over 5 datasets of various sizes

and rendering. When the data size grows past 50,000, rendering has a relative short turn-around of 200 milliseconds, but querying takes up 400 milliseconds.

Interaction logs

The case studies such as the one presented in Case study: Identifying step-up patterns demonstrate how Lifelines2 can be beneficial to analysts in medical scenarios. However, some case studies rely on a set of Lifelines2 features more than the others. For example, in replicating a study that links heart attack incidents to daylight savings time change [29], analysts found the features in temporal summaries in conjunction with *Alignment* are sufficient. *Alignment*, *Rank*, and *Filter* were not necessary. In the Step-Up study, however, more features are required.

By September of 2008, most of Lifelines2 features were complete. Since then, the logging facilities in Lifelines2 had been logging all user actions. The logs keep track of analysts' every action in Lifelines2—*Align*, *Rank*, *Filter*, *Zoom*, *Scroll*, etc. The Lifelines2 log output is in the format of Lifelines2 input, so the logs can be read by Lifelines2 for our analysis. There are a total of 2477 Lifelines2 session logs. However, many of the logs are short, and no case study files were opened aside from the default sample file. These are indicative of testing/debugging sessions instead of analysis/exploration sessions. After removing these testing/debugging sessions, 426 real sessions remain. We loaded these sessions into Lifelines2 for analysis. The temporal summary in Fig. 9 shows the number of events of *Align*, *Rank*, and *Filter*. The minimal amount of activity in January 2009 and summer of 2009 represents winter break and summer vacation. The peculiar spike in October of 2009 represents frequent meetings and analysis of the hospital transfer data with our collaborators. The amount of operations in that period was reflective of the fact that these case studies involved over 15 datasets and many analyses. Table 2 summarizes the logs of the usage of Lifelines2 for the 426 sessions. The operations are broken down into five main categories: ARF, Temporal Summary, Comparison, Data Operations, and Navigation. The table includes the raw number of counts, counts per session, and percentage of sessions that logged such operations.

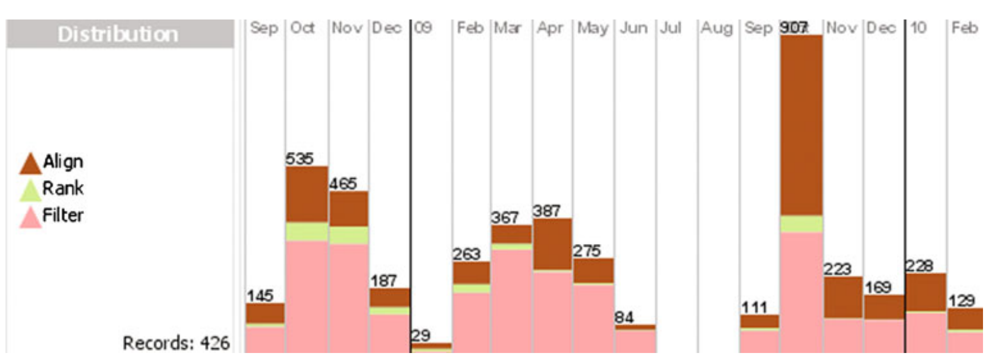


Fig. 9 Lifelines2's temporal summary shows the distribution of *Align*, *Rank*, and *Filter* usage in 426 case studies sessions from September 2008 to February 2010

Table 2 Operator usage in Lifelines2 through our case studies

Operation	Count	Average/session	% sessions
ARF			
Align	1680	3.94	87%
Rank	260	0.61	42%
Filter	2564	6.02	68%
Temporal summary			
Show summary	623	1.46	30%
Temporal selection	531	1.25	24%
Comparison			
Event type change	406	0.95	11%
Comparison type change	79	0.18	12%
Group change	406	0.95	12%
Distribution type change	179	0.42	13%
Data operation			
Keep selected	400	0.94	30%
Remove selected	96	0.23	18%
Save group	409	0.96	29%
Change group	687	1.61	23%
Navigation			
Zoom in	646	1.52	30%
Zoom out	157	0.37	13%
Time range slider	1865	4.38	29%
Change granularity	217	0.51	14%
Scroll	6840	16.10	100%
Collapse	55	0.13	8%
Expand	30	0.07	5%

With respect to the ARF Framework, *Filter* was the most-frequently used operator. *Alignment* was second, and trailed by *Rank*. However, a larger percentage (87%) of sessions recorded at least one use of *Align*, while only 68% had any *Filter*. While *Rank* was useful to reorder the records by their event counts, it was ultimately not a vital operator in our case studies. When pairs of *Align*, *Rank*, and *Filter* were looked at as sequences,

[*Align*, *Filter*] and [*Filter*, *Align*] occurred in 250 (59%) and 211 (50%) sessions respectively. [*Align*, *Rank*] and [*Rank*, *Align*] occurred at 162 (38%) and 123 (29%) sessions respectively. Finally, [*Rank*, *Filter*] occurred in 148 (37%) sessions, and [*Filter*, *Rank*] occurred in only 48 (11%) sessions. When looking at sequences of three operators, the break down (number of sessions that had the contained the sequence) is as follows: *ARF* ([*Align*, *Rank*, *Filter*]): 123, *AFR*: 41, *FAR*: 37, *FRA*: 27, *RFA*: 104, and *RAF*: 87. These numbers indicate that although *Rank* is the least popular of the three operations, when it is used, *Rank* is typically used prior to *Align* or *Filter*, or both.

30% of the sessions used temporal summaries, and 24% used selections in temporal summary. However, the average number of these operations across all sessions was over 1 per session. This means that in sessions that these operations were used, they were used many times, so much that the average count per record is brought up. The operations under the Comparison feature only occurred in 11–13% of all sessions. However, analysts tended to change the event types in the comparison and the groups in the comparison heavily. Changing the type of comparison (Between Group/Within Group/Both) or the type of aggregation (Events/Records/Events Normalized By Record Count) were less frequently used.

In these EHR case studies, analysts tended to use *Keep Selected* as opposed to *Remove Selected* in conjunction with filtering. *Save Group* occurred in 29% of the sessions while *Change Group* occurred only in 23%. This indicates that for some datasets, analysts would save a group, but not change into that group specifically. This situation occurs because Lifelines2 automatically brings the analysts to the newly created group without having them perform the group change themselves. By raw counts, *Change Group*, as expected, is used more frequently than *Save Group*.

The first thing to notice in the navigation operations is that *Scroll* (to pan vertically) is a dominant operation. Every session involved scrolling, and on average, each session has more than 16. Changing the *Time Range Slider* (to zoom or pan horizontally) was a distant second in usage in this category. In contrast, *Change Granularity* (temporal granularity) was not as popular. This may be attributed to the fact that using the *Time Range Slider* analyst can control the temporal range more finely. Even after using the cruder *Change Granularity*, an adjustment in the *Time Range Slider* was often necessary. *Zoom In* was used more often than *Zoom Out*. This is attributed to the fact that users can perform zoom out by using *Change Granularity* or use the *Time Range Slider*. *Collapse Record* and *Expand Record* were the least used features. These features collapse the vertical space of each EHRs so that more can fit in one screen, or expand them to see details more clearly.

A process model for exploring temporal categorical records

Thomas and Cook's defining book on visual analytics, *Illuminating the Path: the Research and Development Agenda for Visual Analytics* [23], presents a canonical process model for analytical reasoning:

1. Gathering information.
2. Re-representing the information to aid analysis.
3. Developing insight through the manipulation of the representation.
4. Producing results from the insight.

This process is repeated as necessary to complete the analyses. We extend this canonical process model and construct one specifically for exploring temporal categorical records by filling out the four steps in detail. We use a multidimensional approach advocated by Shneiderman and Plaisant [22]. The observations of, interviews with, and comments from our collaborators are corroborated with Lifelines2's log data to construct the detailed model. An example sequence of visual analytics steps using Lifelines2 is shown in Fig. 10.

Gathering information

Since Lifelines2 is not directly linked to the databases a hospital may have, we obtain our data through our physician collaborators, facilitated by database administrators. After physicians decide on a medically interesting case study, they begin scoping of the data that they want to examine, e.g. the scope of patients, time frame, relevant events. The physicians then request database administrators of the hospital to gather the requisite data from the EHR system. The de-identified data is then preprocessed into Lifelines2 format for our case studies. At any point of the analytic process, we sometimes revisit this information gathering stage because the physicians (1) become unsatisfied with the data, (2) found systematic errors in the data, or (3) want to incorporate more data for deeper analyses.

In our experience, the information gathering stage typically takes a long time, because of the complexity of the data, underlying data semantics, and infrastructural or organizational barriers. For example, a case study may require data residing in several potentially isolated, databases and medical terms using different IDs and codes in each database. With the paucity of documentation on the mapping of medical terminology to the terms used in the database schema, a specific medical term may be difficult to search, and may require first finding someone who knows where it is. Even with physicians working closely with database administrators, this stage can take from days to weeks, depending on how involved the case study is and the ease in locating the data.

Re-representing information

After de-identification and preprocessing of the raw data to Lifelines2 format, the data is loaded in Lifelines2, and our physician collaborators would examine the result visually in Lifelines2. The visualization enables physicians to better see temporal relationships and investigate common predecessors or successors to a specific event across patients. The physicians would cursorily browse and sometimes examine in detail the data to make sure the data reflects what they know. One of the most common results in seeing data for the first time in a new visualization is the discovery of interesting artifacts such as systematic errors, lack of data consistency, etc. For example, when the data in the mature heparin-induced thrombocytopenia case study was first converted, our physician collaborators found that some patients were given drugs *after* they had been discharged dead! We were

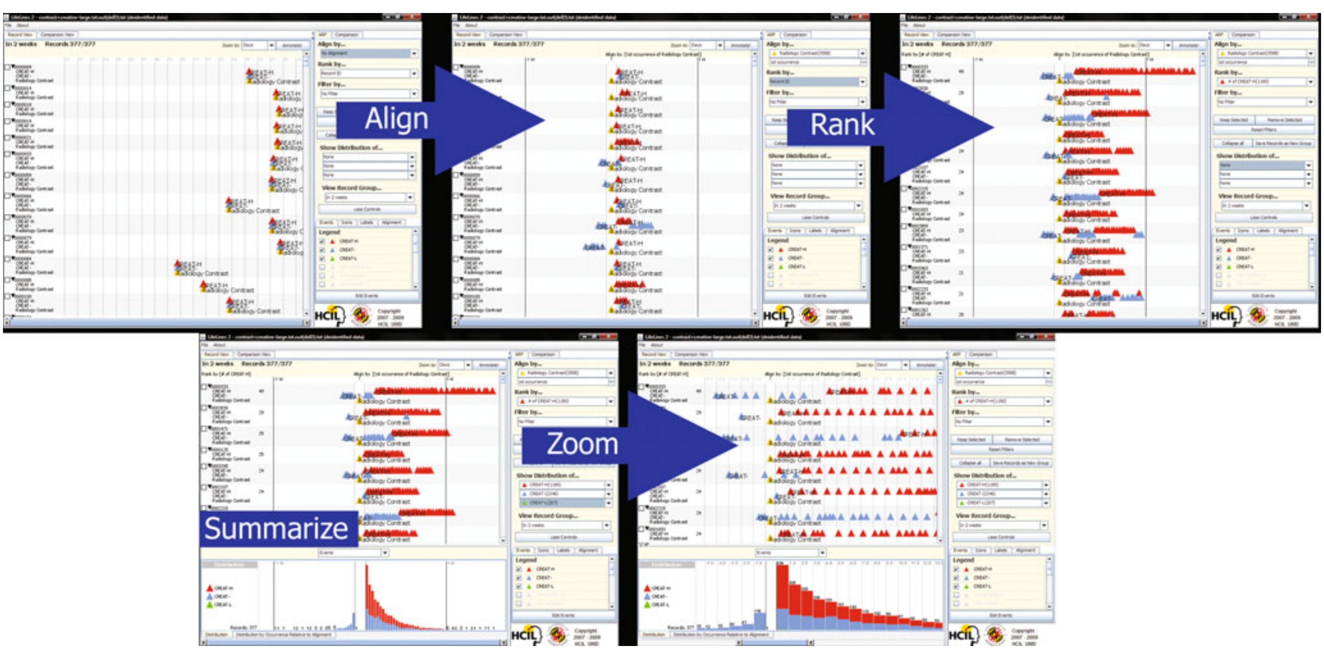


Fig. 10 An example sequence of visual analytics steps using Lifelines2 in the early adoption contrast and creatinine case study. *Alignment* is applied to focus on the orange Contrast events. *Rank* is applied to show patients with the highest number of red Low Creatinine events, and *Temporal Summary* and *Zoom* are used to see details

able to find 7 such cases and determined that they all occurred within 1 h of their *Discharge Dead* events. By consulting the original dataset to make sure this was not an error that occurred in preprocessing, our physician collaborators were able to conclude that this occurred because of the systematic delays in the drug database while the data such as *Patient Discharged* events in other databases are unaffected by the delay. For that case study, this incident raised questions on how reliable the time stamp was for drugs, and whether subsequent case studies would be affected. We eventually found better data to circumvent this particular systematic problem.

Another scenario included inaccuracies in the dataset such as a patient is admitted once, but discharged multiple times. This situation can occur when, for example, multiple databases keep track of the patients discharge status, and when data from these databases are merged naively. Sometimes, however, the data is not usable or is discovered to be unsuitable for a particular case study, and we would take a step back to the data acquisition stage.

Manipulating representation to gain insight

Utilizing different search strategies

After our physician collaborators gain confidence in the data and become familiar with Lifelines2's visualization, they start seeking answers to their questions or finding evidence for their hypotheses. They would change visual representation of the data in order to see event relationships more clearly. This is where visual and data operators such as *Align*, *Rank*, *Filter*, and *Temporal Summary* are used to perform exploratory search. As Table 2 suggests, the order and frequency of each of the operators' occurrence differ case-by-case. In addition, different analysts have different exploratory search style. We have observed that some analysts would apply *Alignment* on different sentinel events in the same exploratory session to look at the data in different views. By using different *Alignment* while showing distribution of certain events they care about in temporal summary, they aimed to find useful or telling "sentinel" events with respect to the events in the temporal summary.

Some analysts take a more traditional approach. They actively manipulate the display by *Aligning*, *Ranking*, *Filtering* iteratively, or changing the temporal summary. When issuing these manipulation operators, analysts would sometimes issue a number of them in succession, and observe the results only at the end of the series of manipulations. This occurs when they are very familiar with the data, perhaps have already gone through a few rounds of analysis. When they are less familiar with the data, they tend to act more tentatively and deliberately. We observed careful examination of the data after each data manipulation operator is applied. They would make meticulous observations on the distribution of events, and comment on whether what they see conform to their understanding of the domain.

Regardless of the strategy they used, *Alignment* remained the strongest indicator on their focus on data. A change in *Alignment* indicates a change of exploratory focus. When the collaborators realize that an *Alignment* would not lead them to the information they seek, they would reformulate the question and subsequently apply a new *Alignment*. This had been observed with different physicians in different case studies, including the mature heparin-induced thrombocytopenia study and the BiPAP study.

Watching results of manipulation

Another important observation of the analysts in exploratory search is that the analysts paid special attention to the change of data at each step of the data manipulation. While this is not specifically mentioned in the canonical process model, we observed this to be a major part of exploratory search in all of our case studies. The physicians would pay close attention to the records on screen. They would also pay attention to how the overall data changes, often by looking at the temporal summaries. Lifelines2 updates the visualization immediately after each of the manipulation operators, and this facilitates our collaborators in identifying the differences between each manipulation step. Table 2 confirms how often the physicians used *Scroll* to view record details. Closer analysis revealed that there are two hot spots where many scroll operations are performed. The first is when the analysts are examining the data for the first time, where the goal is to obtain confidence on the correctness of the data. The second is after the *Align*, *Rank*, and *Filter* operator have been applied. For example, identifying the nuances of sequence *Filters* in the Step-Up case study was accomplished in this manner.

Aside from manual scrolling, our physician collaborators would also keep an eye on (1) the distribution of their favorite events in the temporal summary and (2) the number of records in view out of the total number of records. We observe that when *Align*, *Rank*, *Filter* and *Group* operators are applied, the analysts would focus on these two things. They give the analysts a global feel of how the data is changing when the operators are applied. In fact, in cases where heavy exploration is needed, as in the early adoption heparin-induced thrombocytopenia case study, we notice that the physicians keep their eyes fixed on the temporal summary as a variety of *Filters* are applied. When we work on the mature heparin-induced thrombocytopenia study [28], a different set of physicians also show the same tendency to focus on the temporal summaries as the data is being manipulated. The physicians often ask questions out loud on how a particular manipulation (e.g. *Align*, *Filter*), changes the distribution in the temporal summaries, and they correlate what they see with their experience to better examine if there are unexpected insights. By fixing their attention on the temporal summaries, they gain a good mental model of the data and how the successively applied operators affect the data. They then can decide if their path of exploration seems to be on the right track.

When manipulating representations of the data iteratively, successively, and quickly, the number of representations grows very fast. We anticipated this need and built in some rudimentary mechanisms in Lifelines2 to support resetting and reverting *Filters* and saving subsets of records that can be referred to later on. For example, if physicians do not like a previously applied operator, they can backtrack to a previous state, and rethink their approach. Because our physician collaborators want to keep track of the manipulations they take and understand how these manipulations change the data globally, we have actually observed them combining Lifelines2's features to perform novel tasks that were not initially anticipated. Our physician collaborators would use the grouping operators to create successively smaller groups of patients via Filters and temporal summaries. They would then use the comparison feature to show multiple temporal summaries on these previously created groups to examine if the successive *Filters* seemed to be fruitful, or to determine if a *Filter* is too aggressive. This technique was used in the case studies reported in [28]. Over all, temporal summaries provide indispensable guidance to the physicians. Although focusing on temporal summaries was quick, our collaborators would still examine the records individually when they have the chance, though not exhaustively.

Handling findings

In the process of watching the result of each manipulation, our collaborators are often confronted with a variety of findings. These findings may be a positive one (e.g. one that helps them answer their question), negative (e.g. one that tells them the answer they seek cannot be answered), or unexpected (e.g. an unanticipated characteristic of the data is found, prompting new questions). Regardless which type of finding is reached, our collaborators would, in general, double check their work by revisiting the groups they have previously created and comparing their distributions, or re-performing the data manipulation steps. They would then save their work for dissemination (if it is a positive finding) or for later investigation. Here we present the detailed strategies our collaborators use when each of these types of findings are encountered.

When our collaborators arrive at a point where their questions might be answered, they would use their domain knowledge to comprehend and explain what they see. They first verify how they get to the point by looking at the groups they have created before. They then examine the data in detail to decide whether their questions are answered satisfactorily. If so, then they would take notes and prepare the results for dissemination. More often than not, however, they would find additional, new questions to pursue. When this occurs, we have observed our physician collaborators to utilize their domain knowledge to try to also explain the scenario (e.g. one physician would narrate and reason about why certain EHRs share similar patterns while the others do not) and decide if this is a relevant to pursue. If they are interested in the new question, they would save their current search progress and immediately change the focus to the new question. Alternatively, they would write down new questions for later exploration. We have observed both of these strategies. Depending on the kind of new questions that arise, sometimes it may involve additional data—in which case, the process would loop back to the Information Gathering stage—and sometimes it may only involve using a different set of *Filters* to branch the exploration paths. This has occurred in a number of case studies, including the hematocrit and trauma patient study and the mature heparin-thrombocytopenia study, where discharge status and specific drug prescriptions were later added to investigate new but related questions.

Aside from saving states or jotting down notes, when our collaborators encounter unexpected interesting findings, they would additionally make screenshots and annotate immediately what they see. They would use the built-in screen capture feature in Lifelines2 so they can easily remember what is unexpected and also to share with their colleagues. They would also use the annotation tool in Lifelines2, although annotation was recorded to only occur in 5% of all logged sessions (not listed in Table 2), to annotate the screenshots. If they discovered a set of interesting patients, they would save them and export the results so they can examine the result set of patients in detail, including cross-referencing against their live medical database system.

Unfortunately, sometimes a dead-end is reached. If it is the case that more data is required, or that better preprocessing of the data can help breakthrough the dead-end, we would return to the Information Gathering stage, and restart the analysis. On the other hand, the dead-end can be caused by the limitations of Lifelines2. This occurs when the analysis require features Lifelines2 does not support (e.g., temporal search of numerical values or dealing with patient attributes). Sometimes the limitations of Lifelines2 can be compensated by other systems. For example, in the heart attack and daylight savings case study, we used Excel as a platform to compute average incidents per day. In general, however, unless we find a workaround, the case study would discontinue, such as the situation hematocrit in trauma patient cases study encountered. It is worth noting that none of our mature adoption case studies ran into dead-ends due to Lifelines2's limitations. Over the years, our physician collaborators have become good at picking medical case studies that are appropriate for Lifelines2. This suggests that the features in Lifelines2 facilitate certain ways of thinking, and it takes analysts some time to comprehend and frame their questions appropriately to suit Lifelines2's features.

Producing and disseminating results

Finally, when analysts are able to obtain answers to their questions, they would prepare their findings. Our collaborators routinely keep subsets of EHRs that represent the fruit of their labor, screen shots, annotations, and spreadsheets created in our collaborative exploration sessions. They would additionally ask us to package up the raw data and the final results so their colleagues can verify and examine the analysis results. Our collaborators would then stitch up the results in a presentation to present to colleagues. They show their colleagues or supervisors to argue for or against a procedure/policy change as in the Bounce-Back, Step-Up, and BiPAP case studies. These results can help hospitals monitor the quality of their healthcare, and potentially save operational cost. For example, the results of the Step-Up case study have been presented in at least one physicians' internal meeting, and have been used as evidence to show the step-up rates are normal, and there is no need to change the standard operating procedure of triage.

Summary of the process model

1. Gathering Information
2. Re-representing Information
3. Manipulating Representation to Gain Insight
    a. Utilizing Different Search Strategies
    b. Watching Results of Manipulation
    c. Handling Findings
4. Producing and Disseminating Results

This visual analytics process model is designed to promote insight discovery, based on systematic, yet flexible efforts at data understanding, data cleaning, appropriate representations, hypotheses generation, hypotheses validation, and then extraction of evidence to share with colleagues. This process is iterative, and requires user direction, as it is not yet built into the user interface. This process model is meant to guide users in learning to do discovery in temporal event data and to help designers improve future visual analytic tools.

Recommendations

The case studies, Lifelines2 logs, and observations have revealed some interesting user behaviors when dealing with multiple EHRs. They have also revealed the strengths and weaknesses of Lifelines2. We generalize these into the following seven design recommendations for future developers of visualization tools for multiple EHRs. While some are closely related to Lifelines2, we do try to make them as general as possible.

1. **(Use Alignment)** The usefulness of *Alignment* was evident in the Lifelines2 logs and from observations and collaborator comments. The user logs corroborate the findings of *Alignment* in our previous controlled experiment [27]. When dealing with a large number of EHRs, the ability to use *Alignment* to impose a strict relative time frame was important to our collaborators. It allowed for quicker visual scanning of the data along the *Alignment*. The dynamism of *Alignment* allowed the analysts to quickly switch perspectives and focus if they need to. The idea of "anchoring" the data by data characteristics for exploration had been successful in visualization of other complex data such as network data [34, 35], and *Alignment* seems to be one natural version of it for temporal data. Developing future visualization systems for EHRs should leverage on *Alignment* for its power, flexibility, and wide range of applicability. We would encourage researchers to further explore alternative "anchoring" techniques in temporal visualization.
2. **(Show details)** One surprising finding was that our collaborators liked to look at the details of the records. One piece of evidence is that *Scroll* was the most frequently used operation. Seeing and comparing the details of records seem to reassure the analysts that no data are missing, broken, or lost along the analysis process. Another piece of evidence was that the *Collapse* operator, which makes details harder to see, was hardly used, despite the fact that collapsed records take far less space, and more can fit on a screen. Our second recommendation is that detailed depiction of the records is important, even for multiple EHR visualization, and even if the primary view of the data was to be in an overview, where details may be hidden.
3. **(Overview Differently)** We observed that our collaborators tended to focus on the overview most of the time to get a sense of what each *Filtering* operator does. However, Lifelines2 only provides overview in the form of temporal summaries. Additional concurrent overviews may be beneficial. For example, in addition to the "horizontal" temporal summaries, "vertical" overviews can simultaneously show a different aggre-

gation over records, such as [36, 37]. Furthermore, a good vertical overview design may reduce the amount of *Scroll* necessary, improving overall user performance in information seeking.

4. **(Support Richer Exploration Process)** The features in Lifelines2 that support branching in exploration such as *Save Group* and *Change Group* are, by today's standards, rudimentary. However, they were both used frequently, and we have received comments from our collaborators that a lot of improvements in this regard are desired. As analysis processes becomes more and more involved, analysis tools need to better support branched search, history keeping, and backtracking. An additional requirement for visual analysis systems is to allow users to perform history keeping, backtracking with respect to visualization, not just data. For example, to be able to revert quickly from one representation to a previous one can help users better maintain a consistent mental model.
5. **(Support Flexible Data Types)** Some of the earlier case studies stopped because Lifelines2 does not support numerical values. We discovered that depending on the focus of a medical scenario, sometimes our collaborators reasoned at a higher abstraction (categories), and sometimes lower (numerical values), and sometimes the abstractions change within the same scenario. Most visualization systems focus on either categorical data or numerical data, but there are a few systems that visualize machine-created abstractions [17, 19]. Our experiences suggest that a visualization system that supports temporal analysis seamlessly in multiple, user-driven, and dynamically constructed, abstractions will be valuable.
6. **(Increase Information Density)** The high amount of scrolling we recorded indicates that the amount of data our collaborators want to see is typically much larger than a screen can hold. It is important to improve information density in Lifelines2 and other time-line based visualizations, e.g [13, 16, 38],.. From our experience, the reason our physician collaborators look at the detail of the records is to better understand event sequences across multiple EHRs. To be able to increase information density and preserve the event sequences they care about is very important. Finally, a good "vertical" overview (Recommendation 3) may alleviate this problem at the same time.
7. **(Integration with Live Databases)** Today's clinical information systems contain invaluable information that can be analyzed to monitor and improve healthcare. As [39] suggests, newer medical information systems should contain analytical modules that allow direct connection to the data and provide tools for analysis. We have taken a different route in building and evaluating Lifelines2. We built a visual analysis tool that is agnostic to a underlying storage architecture. As a result, our case studies require considerable amount of efforts to collect, de-identify, preprocess, and convert the data just to begin the analysis. This presents a significant barrier to physician analysts. A tighter integration with a medical information system that has analysis tasks in mind will be better. However, more generalized tools to convert existing data to analyzable data are what will give physicians the power to take the analysis from end-to-end. When designing medical systems, this should be one of the priorities.

While these recommendations are derived from our three-year long design-implement-evaluation process, they are certainly not exhaustive. A different system may derive a different, perhaps overlapping, set of recommendations due to the difference in interaction and visual design, target domain, user, and analytical tasks. While each system can only capture the user experience and analytical styles from its perspective and make recommendations accordingly, we hope our in-depth discussions can foster future dialogues.

Conclusions

We believe the definition of a successful EHR system is not only the storage, retrieval, and exchange of patient data. It should support tasks its end-users care about, and it should be usable and useful. Only then will EHR systems provide value to its end-users and broaden its base of end-users. Collaborating with physicians over the past two and half years, we focused specifically on temporal categorical data analysis tasks. Using Lifelines2, our collaborators were able to make interesting discoveries and help improve patient care. We present a generalization of our eight case studies visualizing EHR data using Lifelines2. By analyzing the feature usage data, user comments, and study observations, we present an visual analytics process model for multiple EHRs and a list of recommendations for future information visualization designers for EHR systems for the tasks of temporal data analysis. While some of our results are limited to capabilities of Lifelines2 applied to EHRs, we were able to draw several other more general recommendations. In this era of vast opportunities for EHR systems, we have made only a small step towards visualization and interface design. We encourage the information visualization designers to continue building a user-centered, task-based design requirements and process models for the betterment of EHR end-users.

Acknowledgements We appreciate the support from NIH-National Cancer Institute grant RC1CA147489-02: Interactive Exploration of Temporal Patterns in Electronic Health Records. We would like to thank MedStar Health for their continued support of our work. We would like to thank Dr. Mark Smith, Dr. Phuong Ho, Mr. David Roseman, Dr. Greg Marchand, and Dr. Vikramjit Mukherjee for their close collaboration.

References

1. Wilcox, L., Morris, D., Tan, D., and Gatewood, J., Designing patient-centric information displays for hospitals. *Proc. of the 28th ACM Int Conf on Human Factors in Comp Syst. 2123–2132*. New York, NY, USA. ACM, 2010.
2. Wilcox, L., Lu, J., Lai, J., Feiner, S., and Jordan, D., Physician-driven management of patient progress notes in an intensive care unit. *Proc. of the 28th ACM Int Conf on Human Factors in Comp Syst*. 1879–1888. New York, NY, USA. ACM, 2010.
3. Zhou, X., Ackerman, M. S., and Zhen, K., Doctors and psychosocial information: Records and reuse in inpatient care. *Proc. of the 28th ACM Int Conf on Human Factors in Comp Syst*. 1767–1776. New York, NY, USA. ACM, 2010.
4. Chen, Y., Documenting transitional information in EMR, *Proc. of the 28th ACM Int Conf on Human Factors in Comp Sys*. 1787–1796. New York, NY, USA. ACM, 2010.
5. Sarcevic, A., "Who's scribing?": Documenting patient encounter during trauma resuscitation, *Proc. of the 28th ACM Int Conf on Human Factors in Comp Syst*. 1899–1908. New York, NY, USA. ACM, 2010.
6. Microsoft., http://www.microsoft.com/amalga/
7. Murphy, S., Mendis, M., Hackett, K., Kuttan, R., Pan, W., Phillips, L., and Gainer, V., et al., Architecture of the open-source clinical research chart from informatics for integrating biology and the bedside. *Proc. Am. Med. Inform. Assoc. Ann. Symp*. 598–602, 2007.
8. Viederman, M., The psychodynamic life narrative: A psychotherapeutic intervention useful in crisis situations. *Psychiater.* 46 (3):236–246, 1983.
9. Greenhalgh, T., Narrative based medicine in an evidence based world. *Brit. Med. J.* 318(7179):323–325.
10. Charon, R., A model for empathy, reflection, profession, and trust. *J. Am. Med. Assoc.* 286(15):1897–1902, 2001.
11. Powsner, S., and Tufte, E., Graphical summary of patient status. *Lancet* 344:386–389, 1994.
12. Plaisant, C., Mushlin, R., Snyder, A., Li, J., Heller, D., and Shneiderman, B., Using visualization to enhance navigation and analysis of patient records. *Proc. Am. Med. Inform Assoc. Ann. Symp*. 76–80, 1998.
13. Bade, R., Schelchweg, S., and Miksch, S., Connecting time-oriented data and information to a coherent interactive visualization. *Proc 22nd ACM Int Conf on Human Factors in Comp Syst*. 105–112. New York, NY, USA, 2004.
14. Pieczkiewicz, D. S., Finkelstine, S. M., and Hertz, M. I., Design and evaluation of a web-based interactive visualization system for lung transplant home monitoring data. *Proc. Am. Med. Inform. Assoc. Ann. Symp*. 598–602, 2007.
15. Horn, W., Popow, C., and Unterasinger, L., Support for fast comprehension of ICU data: Visualization using metaphor graphics. *Method Info. Med.* 40(5):421–424, 2001.
16. Wongsuphasawat, K., and Shneiderman, B., Finding comparable temporal categorical records: A similarity measure with an interactive visualization. *Proc. IEEE Symp. Vis. Ana Sci. and Tech*. 27–34, 2009.
17. Post, A. R., and Harrison, J. H., Protempa: A method for specifying and identifying temporal sequences in retrospective data for patient selection. *J. Am. Med. Inform. Assoc.* 14(5):674–683, 2007.
18. Hinum, K., Miksch, S., Aigner, W., Ohmann, S., Popow, C., Pohl, M., and Rester, M., Gravi++: Interactive information visualization to explore highly structured temporal data. *J. Univ Comp. Sci. – Spec. Iss. Vis. Data Mining* 11(11):1792–1805, 2005.
19. Klimov, D., Shahar, Y., and Taieb-Maimon, M., Intelligent selection and retrieval of multiple time-oriented records. *J. Intel. Inform Syst*. (Published Online), 2009.
20. Chittaro, L., Combi, C., and Trapasso, G., Data mining on temporal data: A visual approach and its clinical applications to hemodialysis. *J. Vis. Lang. Comput.* 14:591–620, 2003.
21. Rind, A., Wang, T. D., Aigner, W., Miksh, S., Wongsuphasawat, K., Plaisant, C., and Shneiderman, B., *Interactive information visualization for exploring and querying electronic health records: A systematic review. Technical Report HCIL-2010-19*, Human-Computer Interaction Lab, University of Maryland at College Park, 2010.
22. Shneiderman, B., and Plaisant, C., Strategies for evaluating information visualization tools: Multi-dimensional in-depth long-term case studies. *Proc. Beyond Time and Errors: Novel Eval. Metho. Inform Vis. Workshop*, 38–43, 2006.
23. Card, S. K., Mackinlay, J. D., and Shneiderman, B., *Readings in information visualization: Using vision to think*. Morgan Kaufmann publishers, San Francisco, 1999.
24. Pirolli, P., and Card, S. K., The Sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. *Proc. Inter. Conf on. Intel. Analy*, 2–4, 2005.
25. Thomas, J. J., and Cook, K. A., *Illuminating the path: The research and development agenda for visual analytics*. National Visualization and Analytics Center, 2004.
26. Hansen, D. L., Rotman, D., Bonsignore, E., Milić-Frayling, N., Rodrigues, E. M., Smith, M., and Shneiderman, B., *Do you know your way to NSA?: A process model for analyzing and visualizing social media data. Technical Report HCIL-2009-17, Human-Computer Interaction Lab*, University of Maryland, 2009.
27. 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. of the 26th ACM Int Conf on Human Factors in Comp Syst*. 457–466. New York, NY, USA. ACM, 2008.
28. 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. Graph.* 15 (6):1049–1056, 2009.
29. Janszky, I., and Ljung, R., Shifts to and from daylight saving time to incidence of myocardial infarction. *New Engl. J. Med* 359 (18):1966–1968, 2008.
30. Wang, T. D., Wongsuphasawat, K., Plaisant, C., and Shneiderman, B., Visual information seeking in multiple electronic health records: Design recommendations and a process model. *Proc 1st ACM Int Health Infor Symp*, 46–55. NY, USA. ACM, 2010.
31. Ventsias, T., Health IT: An Rx for health care. *Terp* 7(1):20–23, 2009.
32. Bederson, B. B., Grosejean, J., and Meyer, J., Toolkit design for interactive structured graphics. *IEEE Trans. Softw. Eng.* 30 (8):535–546, 2004.

33. Wang, T. D., Deshpande, A., and Shneiderman, B. A., Temporal pattern search algorithm, *IEEE Trans Knowl and Data Engin*, Pre-print, http://doi.ieeecomputersociety.org/10.1109/TKDE.2010.257, 2010.
34. Lee, B., Parr, C. C., Plaisant, C., Bederson, B., Veksler, V. D., Gray, W. D., and Kotfila, C., TreePlus: Interactive exploration of networks with enhanced tree layouts. *IEEE Trans. Vis. Comput Graph Spec. Issue Vis. Anal.* 12(6):1414–1426, 2006.
35. Aris, A., and Shneiderman, B., Designing semantic substrates for visual network exploration. *Info. Vis.* 6(4):281–300, 2007.
36. Wongsuphasawat, K., Guerra Gómez, J., Plaisant, C., Wang, T. D., Taieb-Maimon, M., and Shneiderman, B., LifeFlow: Visualizing an overview of event sequences. To appear in *Proc of the 29th ACM Int Conf on Human Factors in Comp Syst*. New York, NY, USA. ACM, 2011.
37. Vrotsou, K., Johansson, J., and Cooper, M., ActiviTree: Interactive visual exploration of sequences in event-based data using graph similarity. *IEEE Trans. Vis. Comput. Graph.* 15(6):945–952, 2009.
38. Stottler Henke. DataMontage., http://www.stottlerhenke.com/datamontage/
39. Patrick, J., and Budd, P., Ockham's razor of design: An heuristic for guiding design and development of a clinical information systems generator. *Proc 1st ACM Int Health Infor Symp*, 18–27. NY, USA. ACM, 2010.