Temporal Search and Replace: A novel tool to simplify event sequences in large complex temporal datasets

Rongjian Lan rongjian.lan@gmail.com PhD CS UMD
Allan Fong af2012@gmail.com PhD CS UMD
Hanseung Lee hansenlie@umd.edu PhD CS UMD
Juan Morales juanmoralesdelolmo@gmail.com Visiting Scholar PhD CS

Fall 2012 CMSC734 Information Visualization Term Project

Abstract
Visualizing and understanding large complex temporal datasets can be difficult and confusing because of the structure of events and event sequences. We introduce a novel Temporal Search and Replace tool that lets users easily and intuitively group and simplify related events sequences or repeated event patterns. The features in the tool were evaluated with a usability study of nine participants. Overall, the tool was very helpful at reducing visual clutter and in improving the participants’ understanding of the data. Challenges and recommendations for improving the tool are also discussed.

Keywords: Temporal search and replace, simplification, event sequence, grouping, visual analytics, information visualization

1. Introduction
Visualizing and understanding the aggregation of multiple temporal events is difficult and complex often because of how the events are categorized, sequenced, and aggregated. The complexity with exploring and understanding very large temporal datasets can easily overwhelm a user’s perceptual and cognitive capabilities. This is especially true for users interested in exploring sequences of events, rather than individual events.

Although our minds group and cluster related temporal events to simplify temporal information, very few temporal information visualization systems let users simplify events sequences and patterns in a way that is intuitive and user-defined. In this paper, we introduce a novel Temporal Search and Replace tool that simplifies large complex temporal datasets with user-defined event sequences and replacement glyphs. This feature is integrated into the existing EventFlow software, leveraging EventFlow’s search capabilities [Monroe 2012].

Our contribution gives users a new way to simplify large complex temporal datasets into meaningful groups or clusters. A usability study showed how participants were able to use the Temporal Search and Replace tool to reduce information clutter and improve their overall understanding of their data. We also compiled and ranked a list of improvements for our tool based on the recommendations and feedback we collected during the usability study.

2. Search and Replace
Search and Replace (sometimes referred to as Find and Replace) tools are widely utilized in many software tools, such as Microsoft Word and other text editing software. Search and Replace can help users rapidly process large documents or datasets to find information and propagate changes. These Search and Replace tools are very beneficial and can be easily implemented with various linear search algorithms. Although most of the Search and Replace tools used today search sequential primitive data types (characters, integers, etc.), information stored in other data types and data structures, such as graphical and temporal information, can also benefit greatly from search and replace capabilities.

Searching and replacing images and illustrations is complicated and an area of active research [Chok
Kurlander and Feiner were amongst the first to demonstrate the capability to search for graphical constraints and replace the results with new illustrations [Kurlander, 1992]. Effectively searching and clustering graphical images can also benefit the human computer interaction as demonstrated with Sikuli [Yeh 2009]. Sikuli can search a screenshot of a Graphical User Interface (GUI) and automate the functionality of that GUI with its pictorial representation.

Applying search and replace capabilities can also help users simplify and understand large complex temporal dataset. However, there is little research in the area of user-defined search and replace capabilities for event sequences. While researchers have proposed various visualization strategies to simplify temporal data, these visualizations do not easily lend themselves to the identification and simplification or replacement of user-defined event sequences and patterns [Wongsuphasawat 2011, 2012][Luo 2012][Havre 2002][Dongning 2012]. Lee et. al. proposed using a temporal event matrix representation to mine temporal data for event patterns [Lee 2011]. Researchers have also proposed graphical queries of time series and many other ways to search for temporal patterns [Wattenberg 2001][Koch 2011][Haigh 2004]. While these techniques are beneficial for exploratory analytics, they still have limited flexibility in letting users define patterns and sequences of interest. Our Temporal Search and Replace tool provides users this capability of grouping and simplifying related events, which is reflective of our natural tendency in processing and recalling related temporal events. Better understanding of how our mind process temporal information helps us design more intuitive interactions and visualizations.

3. Simplification of Events in Memory
We have a tendency to group and cluster related information with each other to simplify the information presented to us [Farrell 2012]. This temporal grouping and clustering can help us recall past events by simplifying how we remember and store our temporal memories [Farrell 2012]. Researchers have also shown the potential benefits of “chunking,” a term used in psychology, as an extension of Miller’s magic number of seven plus or minus two summarized by [Miller 1956]. Remembering telephone numbers is a good example of chunking. Telephone numbers are comprised of ten digits. Rote memorization of ten digits can be difficult. However, the first three digits are usually grouped together, or chunked, as an area code. Therefore, instead of remembering three extra digits, one just needs to associate one location reference with the telephone number’s area code. Gobet et. al. further summarizes research supporting the benefits of chunking [Gobet 2001]. The benefits of grouping related information also pertains to temporal data. Farrell highlights a growing body of work suggesting that “we naturally parse our experience into discrete, temporally extended events... and that this parsing has effects on memory performance” [Farrell 2012].

Furthermore, studies have shown that we typically group and store similar events together, reflecting our underlying tendencies to group and simplify temporal data. For example, it will be difficult to distinguish one’s birthday ten years ago from one’s birthday eleven years ago. As long as nothing out of the ordinary happened between the two birthdays, our minds will most likely simplify those two distinct memories into one. It is intuitive for us to simplify temporal events whether through grouping, clustering or simplifying repeating events. Our Temporal Search and Replace tool aims to complement this natural tendency by letting users group and simplify events while maintaining the specific details for each event so users can filter, zoom, acquire details on demand. (*Please note we are will be working with Professor Norman (Psychology) until early next week, 12/4, to better elaborate and cite this sections.)

4. Simplification of Event Patterns
The Temporal Search and Replace tool helps users simplify large complex temporal datasets by giving users the flexibility to define, search, and replace event sequences and patterns. For example, users can group related events together and represent the sequence in one glyph. Users can also highlight repeating patterns or search for sequences using wildcard notations. We built our tool in EventFlow because
EventFlow already has a working platform for visualizing temporal point and interval events. We were also able to leverage EventFlow’s GUI search Application Programming Interface (API) for our Temporal Search and Replace tool.

4.1 Point and Interval Events
Our tool handles two types of events: point and interval events. Point events are defined by one time stamp while interval events are defined by two time stamps, a beginning time and an end time. Although these event types are similar and interchangeable (interval events can be transformed into two point events and point events are rarely instantaneous, without a duration), in practice, it might sometimes be more helpful to consider an event as a point event, and other times, it might be better to consider the same event as an interval event. Our tool provides users the flexibility to translate between point and interval events.

4.2 Defining Search and Replace Rules
We worked with Catherine Plaisant and Megan Monroe from the Human Computer Interaction Lab at the University of Maryland College Park, on understanding user requirements for the Search and Replace features. Together we brainstormed designs and useful features that can ultimately be integrated into EventFlow. We also met with Sheila Weiss, from the University of Maryland Baltimore Medical Center, twice, as a potential case study opportunity. However, there was data restrictions issues that slowed down the process. Nevertheless, Sheila provided helpful real-world perspectives on the needs and usefulness of our tool and we hope to continue collaborating with her in the future. The following is a list of the search and replacement rules implemented by our Temporal Search and Replace tool.

Replacing one event with another event
This replacement feature lets users form groups or clusters of similar events based on user-defined attributes. For example, patients given the same drug but with different dosages are sometimes considered different point events. Other times, it would be more convenient to consider them the same event. This replacement feature lets users easily select appropriate groupings and visualizations for their analysis.

Replacing one event with many events
An example of replacing one event with multiple events can be transforming an interval event into two point events, a start and end date. This replacement can sometimes simplify overlapping intervals to reduce visual clutter.

Replacing many events with one event
This is probably the most common type of event simplification, where users replace many events with one event. For example, if researchers are interested in the time between a high blood count point event and a low blood count point event, they can replace the two point events with just one interval event.

Replacing many events with many events
This simplification rule would be useful for users interested in simplifying multiple events while still maintaining specific features of the original sequence. For example, researchers might only be interested in the duration of a patient’s hospital stay and when the patient got discharged from the emergency room. By replacing their search sequence with one interval and one point event, they could easily remove extraneous hospital events from the visualization.

Repetition
The repetition option lets users search and replace all the occurrences of a user-defined sequence both within and between entries. Events patterns may occur multiple times in a single record. For example, patients could routinely take three drugs (Drug A, Drug B, and Drug C) weekly. Visualizing all these
three distinct point events can be distracting and add to the visual clutter on the display. Our repetition feature can simplify this visualization by searching and replacing the repeated events within a record.

Order
Sometimes users would be interested in the occurrence of multiple events without any constraint for their ordering. As a result, we implemented an option in our Search and Replace tool that lets users specify whether or not the order of the events in a sequence should be considered in the search.

Wildcard
Lastly we implemented a point event wildcard and an interval event wildcard search features. Sometimes users would be interested in Event A and Event B only if there was another event, any event, in between the two. For example, researchers could be interested in patients who went to the Emergency Room (ER) and Intensive Care Unit (ICU) only if there was another event (a stroke event, a drug delivery event, etc.) in between the ER and ICU events. Our wildcard notation enhances the current EventFlow search capabilities, giving users more flexibility in their searches.

5. Event Data Structure Description
The original EventFlow event sequence searching algorithm only support exact sequence matching, which means the query sequence targets one-to-one match of a fix-sized target sequence. To enable the searching algorithm to match a more broad range of patterns, we extend the original EventFlow searching algorithm to support repetition, no order and wildcard constraints. While the original searching algorithm store the matched sequence in an array of fixed length, we use a resizable list to keep track of the matched sequence. Also, we need to remember how the current matched event corresponds to the elements in query sequence so that the backtracking process [Wang 2009] works as expected. In addition, since our tool supports the hierarchical constraint specification (e.g. a no order constraint inside a repetition constraint or nested constraints), a tree structure is used to represent the constraint hierarchy and a stack is used to store the active constraints during the searching process.

After the searching algorithm finds all the target sequences in each record, we store them as matched events within that record. This is necessary for our replacement operations. During the replacement operation, the matched events are scanned and the new events of replacement are created based on the alignment information. Finally, the matched events are hidden and the new replacement events are added to the record.

6. Search and Replace User Interface Description
Users can search point events, interval events, and wildcard events the same way as they would in EventFlow. Figure 1 shows an overview of the EventFlow interface. In addition, users can set repetition and no order constraints by selecting the events in the search panel via mouse dragging as shown in Figure 2 and Figure 3. To support repetition and no order efficiently, we maintain a hierarchical tree data structure to store the constraints, as previously discussed. This tree data structure is used to efficiently search for patterns as well as efficiently visualizing constraints without overlaps.
The replacing process starts by clicking on the “Replace As” button on the search panel. As shown in Figure 4, a panel pops up visualizing the sequence pattern users searched for and the sequence pattern users want to replace to. Now users can interactively add existing point events, existing interval events, new point events, and new interval events to the replacement sequence, Figure 5. If users add new points events and new interval events, it is added to the existing event list so that users can use these events again. The added events are automatically aligned to the nearest event marker in the original sequence.
7. Usability Study
To evaluate the features in our Temporal Search and Replace tool we conducted a usability study. We recruited 9 participants (5 males, 4 females, mean age 26.7, std 2.5) all who frequently users of computers, have previously used Find and Replace software functions, and have basic understanding of the difference between point and interval events. The involvement of these participants were all voluntary. Our usability study was composed of four main phases: background review, training, usability testing, questionnaire and interview.

7.1 Background Review
We reviewed basic temporal concepts relevant to our study. We also explained the layout of EventFlow and general description of the Search and Replace features. We provided participants handouts to guide our discussion. The handout material for the background review, training, and testing can be found in the appendix.

Training
During the training section, we walked through six tasks with each participants. A training dataset was used to help participants understand and learn the interface. We used a handout again to aid in this process. The participants were able to take notes on the handouts if they wanted. We encouraged the participants to think-out-loud, and to ask questions and comment on features or concepts they don’t understand. We recorded and address these questions and comments.

Usability testing
After the training session, we loaded a new simulated dataset we generated of 40 professors for the participants and provided the participants another handout of tasks and questions. The dataset contains 7 unique point events (when the professor received a Bachelors, Masters and PhD degrees, journal and conference publication dates and, if applicable, dates of book and newspaper publications). The dataset also contains 3 unique interval events (the time period the person spent as an assistant professor, as an associate professor, and as a full professor). The tasks and questions were very similar to the training tasks and questions:
1. Search for event sequences
2. Replace search results with new interval event
3. Search for people who had both book and newspaper publications where order of the publications is not important
4. Search for people who have at least 10 conference publications
5. Search for and replace all journal, conference, book, and newspaper publications with a new point event

We encouraged the participants to think-out-loud, comment on anything they didn’t understand, find difficult, and so on. Our involved during this phase was limited to technical software issues or to help with additional clarification and guidance if participants started to struggle. We encouraged the participants to think and talk through each question. We documented their questions, comments, and problems that arose. After completing the usability tasks, we asked the participants to fill out a survey and answer follow up interview questions. These questions as can be found in the appendix.

8. Survey Results

*Very noticeable reduction in visual clutter*

The results show that, for the participants, the reduction in visual clutter after using the Temporal Search and Replace tool was very noticeable. The mean result was a 5.4 on a 1-7 point Likert scale where 7 is being extremely noticeable, see Figure 6.

*Very helpful in improving understanding*

Furthermore, the results show that using the Temporal Search and Replace tool was very helpful in improving the understanding of the participants. The mean result was a 5.0 on a 1-7 point Likert scale where 7 is greatly improved understanding, see Figure 6.

*Recall difficulties*

The mean result for easy of recalling what each event simplification represented was a 4.3 on a 1-7 point Likert scale where 7 is being extremely easy to remember, see Figure 6. This highlights some difficulties in remembering what the different replacement glyphs represented.

![Figure 6: Averaged responses for the utility of the tool at reducing clutter, improving understanding, and helping recall the simplified pattern. In general, 1 is not at all helpful or could not remember at all. Standard error bars are included.](image)

*Features in the Temporal Search and Replace tool*

Figure 7 shows the average results of how intuitive the No Order, Repetition, and Wildcard features were in the Temporal Search and Replace tool. Results were measured using a 1-7 point Likert scale.
where 1 is not at all intuitive and 7 is extremely intuitive. Both the concept and implementation of the features were assessed. In general, the more difficult a feature was to understand (the lower the concept score), the harder it was to use (the lower the implementation score). Furthermore, the repetition feature was both the easiest to understand and to use.

![Figure 7: Average responses for the level of intuitiveness of both the concept and implementation of the No Order, Repetition, and Wildcard features. 7 is extremely intuitive and 1 is not at all intuitive. Standard error bars are included.](image)

### 9. Discussion
The usability study demonstrated the overall utility of the Temporal Search and Replace tool, especially at reducing visual clutter and helping improve understanding of complicated temporal datasets. This was reflected in both the survey results and comments from participants. For example, five of the nine participants specifically commented that it was easy and intuitive to do a basic search and replace, especially replacing multiple events with one interval event. Our results also showed that the repetition search feature was more intuitive than both the no order and wildcard features. Nevertheless, all the features were generally easy to understand.

### 9.1 Integration with EventFlow
Building and integrating the Temporal Search and Replace tool into the current EventFlow system was advantageous but also challenging. By integrating with EventFlow, we were able to take advantage of many existing visualization techniques and APIs. However, it also meant that we had to spend some time combing through and understanding the EventFlow code, especially during debugging. Furthermore, it is important to note that EventFlow offers many capabilities we did not tell our participants about. There are EventFlow features that could have helped participants in the usability study. However, we wanted to very narrowly scope our study to focus on evaluating the Temporal Search and Replace tool. As a result, some participants made comments about seeing summarized matched records or ranking features because we didn’t cover these features in the training.

### 9.2 Challenges and Areas for Improvement
Despite the positive feedback of our tool, there were many challenges and difficulties that emerged. We categorized and prioritized (by impact and ease of implementation) the challenges that we noticed, the difficulties participants expressed and their suggestions for improvements. Impact is ranked as low, medium and high. Ease of implementation is ranked as hard, medium, and easy. These challenges and recommended improvements are summarized below.
Several participants had difficulties using the features in the Replace As panel. First, it was not intuitive to the participants that they had to click to add an event sequence. Furthermore, if they added an interval, it was not intuitive that they could modify the interval by extending the interval marker. Two participants also had a lot of difficulties with selecting on the small interval end markers. One participant also mentioned how the spacing between the two event sequence boxes was too large, making it difficult to vertically aligning events. Furthermore, we received comments on how user controlled incremental replacement of events could be helpful.

**Recommendations**

R1. Set a default replacement glyph for users that they can easily edit  
   (impact – high, implementation – easy)

R2. Set default interval glyphs to span the entire search event sequence  
   (impact – high, implementation – easy)

R3. Make it easier to select and drag edges of interval markers  
   (impact – medium, implementation – medium)

R4. Reduce space between two event sequence boxes  
   (impact – high, implementation – easy)

R5. Add visualization of where replacement events are snapping to in the original sequence  
   (impact – medium, implementation – high)

R6. Add features to let users search and incrementally replace sequences  
   (impact – high, implementation – high)

**Unnecessary clicks**

Two participants suggested reducing the number clicks while using the tool, particularly the extra steps to search first before replacing. They mentioned that this was not value added and could be happening behind the scenes.

R7. Combine the two-steps search and replace process into one step  
   (impact – medium, implementation – easy)

R8. Map the flow of user clicks for all tasks to systematically eliminate unnecessary clicks  
   (impact – high, implementation – high)

**Event constraints**

One participant mentioned that accessing the event constraints window was not intuitive. These features are completely hidden until the users calls it. Furthermore, sometimes the event constraint window was accidentally activated because the participant would slightly move the mouse after clicking.

R9. Activate event constraint windows with a button rather than the click and draft option  
   (impact – medium, implementation – high)

**Common wildcard**

Participants also expressed the general need to have a generic wildcard expression, not just a point event wildcard and an interval event wildcard.

R10. Create a generic wildcard  
   (impact – medium, implementation – high)

**Additional improvements**

We received many additional suggestions for ways to further improve our tool.
**R11.** Include summary of the number of replacements found  
*impact – high, implementation – medium*

**R12:** Add time constraints for event sequences, such as searching between two time periods  
*impact – high, implementation – high*

**R13:** Make interval glyph more representative of intervals, particularly in the overview panel  
*impact – low, implementation – medium*

**R14:** Include additional ranking features, such as ranking by interval duration or ranking by the number of events in an interval  
*impact – high, implementation – high*

**R15:** Automate the alignment of event sequences after a search  
*impact – low, implementation – medium*

### 9.3 Design Improvements
After our usability study, we were able to make a few improvements to our tool based on feedback we received from the participants. We were able to change a number of naming conventions to make our labels more consistent and understandable. Furthermore, we improved the pattern visualization summary panel to help users remember their simplification patterns.

### 9.4 Future Work
Given the outcomes of the usability study, we will implement the Temporal Search and Replace tool recommendations focusing first on those that are high impact and easy to implement (R1, R2, R4), then focusing on the remaining high impact recommendations (R6, R8, R11, R12, R14). We will then work on implementing the more difficult fixes with less impact, such as making a generic wildcard event. Furthermore, we will review the relevant EventFlow recommendations with the EventFlow development team to determine their applicability and to develop a plan for implementation.

To further evaluate our tool, we will conduct a long term case study with Sheila Weiss, a clinician and research at the University of Maryland Baltimore medical center. Given more time, we will also repeat the usability study with more participants or conduct a controlled experiment to collect quantitative metrics, such as response time and accuracy, to further assess the utility of the Temporal Search and Replace tool.

### 10. Conclusion
Large complex temporal datasets can be difficult to simplify and understand. In this paper, we introduced a novel Temporal Search and Replace tool to help users group and simplify related events into sequences or repeating patterns. We implemented and evaluated several search and replacement rules in a usability study. The study highlighted several challenges and recommendations for improvements of our tool. However, the participants, overall, found the tool to be very helpful at reducing visual clutter and improving understanding of complex temporal datasets.

### 11. Credits
Rongjian Lan - Team lead, Data structures design and programming lead  
Hanseung Lee - User Interface & Interaction design and programming lead  
Allan Fong - Usability design and Technical report writing/Video lead

### 12. Acknowledgements
This work would not have been possible without the guidance and support of Catherine Plaisant and Megan Monroe. Catherine and Megan were extremely helpful in helping us understand the requirements, constraints, and rules for our tool. Megan was also very helpful in our understanding of the current EventFlow program, which was very important when it came to integrating with the system. Furthermore, we would like to acknowledge Sheila Weiss in helping us understand how our feature could be beneficial for looking at medical records and providing us feedback on practical features that would be important.

13. References


Oro, E., M. Ruffolo, and S. Staab. SXPath: Extending XPath towards spatial querying on web


