Visual Analysis of Massive Web Session Data

Zeqian Shen* eBay Research Labs
Jishang Wei† University of California, Davis
Neel Sundaresan‡ eBay Research Labs
Kwan-Liu Ma§ University of California, Davis

ABSTRACT

Tracking and recording users' browsing behaviors on the web down to individual mouse clicks can create massive web session logs. While such web session data contains valuable information about user behaviors, the ever-increasing data size has placed a big challenge to analyzing and visualizing the data. An efficient data analysis framework requires both powerful computational analysis and interactive visualization. Following the visual analytics mantra "Analyze first, show the important, zoom, filter and analyze further, details on demand", we introduce a two-tier visual analysis system, *TrailExplorer2*, to discover knowledge from massive log data. The system supports a visual analysis process iterating between two steps: querying web sessions and visually analyzing the retrieved data. The query happens at the lower tier where terabytes of web session data are processed in a cluster. At the upper tier, the extracted web sessions with much smaller scale are visualized on a personal computer for interactive exploration. Our system visualizes a sorted list of web sessions' temporal patterns and enables data exploration at different levels of details. The query-visualization-exploration process iterates until a satisfactory conclusion is achieved. We present two case studies of *TrailExplorer2* using real world session data from eBay to demonstrate the system's effectiveness.

Index Terms: H.5.m [Information Interfaces and presentation (e.g., HCI)]: Miscellaneous—

1 INTRODUCTION

Providing frictionless user experience is the key to the success of a website. As a prerequisite, comprehensive understanding of the user experience is crucial for the website to improve service. Usually, a user browses the website, leaving a trail of which pages he visited and how long he stayed on each page. Such time-stamped event sequences, recorded as web sessions, largely reflect users' feelings about the website service. Analyzing these event sequences can highly advance the understanding of user experience.

Visual analysis has emerged as a powerful technique to discover knowledge from data. Visual analysis combines data analysis methods with interactive visualization to enable comprehensive data exploration. Compared with the conventional statistical methods, the visual analysis approach is data-driven and exploratory rather than hypothesis-driven and confirmatory. It well fits in a situation where analysts have little knowledge about data. In addition, visual analysis fully takes advantage of human perceptual and reasoning abilities to carry out thorough data analysis at both the overview and detailed levels. The visual analysis methodology has been widely adopted to deal with web session data.

Despite of the popularity of visual analytics, the sheer data size and complexity nature of web sessions place a big challenge to the visual analysis system design. On one hand, the website relentlessly collects web session logs accumulating to a huge volume of data resources. On the other hand, the event ordering and how long each event lasts are the focuses of analyzing the web sessions. Determining how to visually encode each individual time-stamped event sequence and how to classify the event sequences to indicate group behaviors are non-trial. Intuitive visual representation and interaction are needed to facilitate the analysts' perception and reasoning.

In our work, we created a two-tier visual analysis system, *TrailExplorer2*, to address the above challenges. One tier is a large-scale Hadoop-based data query engine to extract data of interest; the other is an interactive visual interface to support visual exploration. Following the visual analytics mantra [3] by Keim, "Analyze first, show the important, zoom, filter and analyze further, details on demand". A typical data investigation scenario starts from "analyze first" by launching a query across the massive data. After the data of interest are extracted and fit in a computer memory, the user interface "shows the important" by visualization. A range of interaction toolkits facilitates data examination. Whenever the analyst requires details, the summary statistics for a specific chosen session is presented.

In sum, our work contributes to the visual analysis of large-scale web session data in three aspects,

- Designing a visual analysis system by integrating Hadoop-based data query techniques and interactive visualization.
- Devising an in-memory tree structure to support levels-of-detail visualization.
- Inventing intuitive user interaction techniques to enable thorough data exploration at both the overview and detailed levels

In the rest of the paper, we give a brief background of the web sessions analysis, followed by an overview of our two-tier visual analysis system. We then present two real-world use cases on eBay's web session data. Both cases show our system's usefulness for providing actionable insights to the business.

2 RELATED WORK

Regarding the large data size and complexity nature of time-stamped event sequences, in this section, we review how the existing work has made efforts to approach those problems.

2.1 Large-Scale Data Exploration

Shneiderman [10] first introduced the information seeking mantra, "Overview first, zoom and filter, then details on demand" for data exploration. Following this guideline, if we can design systems with authentic data overviews, intuitive interaction techniques and effective data analysis methods, users can take on even more ambitious data analysis tasks. However, researchers become reluctant to adopt this design guideline when solving big data problems. One reason is that it is impractical to generate an overview of the massive sessions on a constrained display. Even irrespective of the display constraint, humans' perceptual and cognitive ability would quickly reach limitation given a large amount of data. Another reason is that it is unnecessary to provide an overview in many cases where the data summarization doesn't relate to the main data analysis goal.

*e-mail: zeqshen@ebay.com
†e-mail: jswei@ucdavis.edu
‡e-mail:nsundaresan@ebay.com
§e-mail:ma@cs.ucdavis.edu

IEEE Symposium on Large Data Analysis and Visualization 2012
October 14 - 15, Seattle, Washington, USA
978-1-4673-4733-4/12/$31.00 ©2012 IEEE

Considering the difficulties in analyzing the large data, Keim extended the information seeking mantra to the visual analytics mantra [3], "Analyze first, show the important, zoom, filter and analyze further, details on demand". The new mantra introduces a computational analysis step that reduces size and/or complexity of the data before trying to visualize it. Following Keim's visual analytics mantra, Andrienko and Andrienko [1] presented a systematic design of a toolkit that could support visual exploration and analysis of massive collections of movement data. The visual analysis process is iterative and involves three major steps: computational analysis, visualization of the computational results, and interactive visual analysis. Similarly, Zhang and You proposed a dynamic tiled map services approach [18] to support visual explorations of large-scale raster geospatial data in a web environment. Their method allows for querying the interesting portion of data and then visualizes the interesting data.

There are many other alternative principles to the information seeking mantra targeting on analyzing and visualizing large-scale data sets. The principles exclusively suggest a framework with a preprocessing step and then visualization. Frank van Ham and Adam Perer [12] advocated the "Search, Show Context, Expand on Demand" model to support large graph exploration with degree-of-interest. Their system allows users to start data exploration by searching an interesting point and shows the context. Users can remotely browse the immediate context graph around a specific node of interest. *TreePlus* [6] supports exploration of the local structure of the graph and gathering of information from the extensive reading of labels using a guiding metaphor of "Plant a seed and watch it grow". It allows users to start with a node and expand the graph as needed.

2.2 Visual Analysis of Time-Stamped Event Sequences

Web sessions, electronic health records and accident response logs are all examples of time-stamped sequences of events. Extensive work has been done to visually analyze the sequence data across a vast range of application areas. Schaefer et al. [8] introduced a visualization method using ordered color rectangles to support identifying significant events, event sequence patterns and event clusters. Krstajic et al. [4] developed *CloudLines* to visualize multiple time series event data. *LifeLines* [7] is a visualization system that presents personal histories, including medical and court records, professional histories and other types of biographical data. Wang et al. designed an interactive visualization tool, *LifeLine2* [13, 14], to visualize and analyze temporal categorical data, especially electronic health records. Wongsuphasawat et al. introduced *LifeFlow* [16], an interactive visual overview of event sequences, to analyze event sequences in the medical and transportation research areas. Ahn et al. [17] applied *LifeFlow* to analyze user behavior patterns of the adaptive exploratory search systems. Similar to the previous work, we visualize the time-stamped sequences using time-line bar chart. However, besides data visualization, we focus on providing a systematic solution to analyze the large-scale event sequences. Our visual analysis system is based on a previous prototype system, *TrailExplorer* [9]. Shen and Sundaresan introduced this prototype to analyze temporal user behavior patterns in webpage flows. *TrailExplorer* handles preprocessed web session data of a manageable size. Therefore, the visual exploration is confined by the limited data. In contrast, *TrailExplorer2* runs directly on the massive raw session data by integrating a Hadoop-based data query platform. In addition, useful interactions, e.g., aligning sessions by arbitrary events, are added to enable more advanced analysis.

3 Visual Analysis System Design

In this section, we discuss how analysts at eBay study the web sessions in their work. The experience learned from real-life analysis processes illustrates how we should design a visual analysis system to facilitate data exploration.

3.1 Web Session Data

A web session is a time-stamped sequence of events [5]. An event corresponds to a user action, such as clicking a button or filling a form. Every event has a time stamp, which records when the event happened. Based on the time stamps, we can know the ordering of the events. The elapsed time is calculated as the difference between the timestamps of two consecutive events. It indicates how much time a user spent at each event, and the ordering shows the path a user follows. Particularly, the events before and after a designated event are called the inbound and outbound respectively, which is mostly an analysis focus. Usually, there is one event defined as a completion event. A web session with the completion event coming last is considered as a completed session. The percentage of sessions that are completed, regarded as the success rate, is an important metric in the web session analysis.

3.2 How Analysts Study Web Session Data

By taking the chance to work with eBay analysts, we observe how they analyze the web session logs. We summarize four typical analysis scenarios as follows. These real-life analysis scenarios inspire us in dealing with the problems of large-scale web session visual analysis.

- Analysts try to find the correlation between different event sequence patterns and the success rate. Most of the time, they form a configuration of an event sequence pattern, query all similar event sequences, and investigate the success rate for the sequence pattern.

- Analysts check the elapsed time of each event and examine its correlation with the success rate. If a long lasting event causes dramatic decrease of the success rate, analysts would look into this particular event to find out causes.

- Analysts study the elapsed time of one step, inbound and outbound of the specific event, the success rate of a specific event sequence pattern, together with statistical results.

- Analysts conduct the analysis iteratively. They query an event sequence pattern to testify whether a hypothesis is right. If not, they might form new hypothesis to study further.

In a word, the web session analysis is an iterative process as illustrated in Figure 1. It involves querying user-defined event sequence patterns, investigating how the ordering of events, the elapsed time at each event and the associated statistics correlate to a goal metric, such as the success rate.

3.3 System Design Principles

One big challenge of analyzing massive data is the sheer data size. From the discussion in Section 3.2, we notice that the real-life data analysis mostly focuses on the data of interest. Therefore, if our system allows analysts to retrieve interesting data on request, we don't need to run analysis on the entire data set every time. It will highly reduce the data size for visual analysis. However, even only handling a subset of the entire data, in most cases, the data size is still fairly big. We need to aggregate the data to further reduce the size in order to fit them in the memory of a personal computer for real-time interaction.

Regarding the data complexity, two critical aspects of web sessions are the events ordering and the elapsed time of events. The system ought to emphasize on delivering these data characteristics. In addition, the system should support levels-of-details abstraction so as to accommodate scalable data visualization, and provide interaction tools, such as zooming and selection, to enable free data

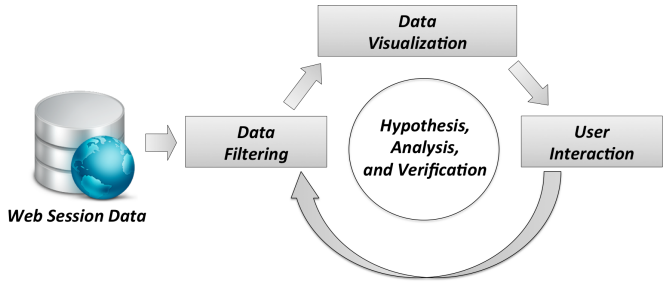

Figure 1: An iterative data analysis process includes three main steps: retrieving the data of interest, visualizing the retrieved web sessions, and exploring the data features.

exploration. We summarize the system design principles in three aspects:

- Support data query to extract temporal event sequences according to a user-defined sequence pattern.
- Demonstrate both the ordering and elapsed time of the event sequences.
- Provide interaction to examine any event and their neighbors, perceive data at both overview and detailed levels and check details of each temporal event sequence pattern.

4 Visual Analysis System Implementation

Our visual analysis system, *TrailExplorer2*, is two-tier. One tier is a Hadoop-based data query engine to extract data of interest; the other is an interface to provide intuitive visualization and interaction.

4.1 Data Filtering

When *TrailExplorer2* first starts, users are asked to select a set of events of their interests. The system then finds all the sessions that contain any of these events. The data extraction contains all distinct event sequences and their number of occurrences in the data. At eBay an analysis is often conducted on terabytes of data which contain hundreds of millions sessions. Based on our interviews and surveys with analysts, their preferred average waiting time between sending requests and having the data ready for exploring is below 10 minutes. Therefore a scalable data processing platform that can perform the filtering on such vast amount of data is desired.

MapReduce [2] is a prevalent framework to handle huge datasets using a cluster of computer nodes. The data processing involves two steps:

- "Map" step: The input data are split into chunks which are processed by map nodes in a completely parallel manner.
- "Reduce" step: The outputs from the map nodes are collected and combined in some way to form the final answer.

Our data query is a typical pattern-based searching, which is highly distributable and can be solved very efficiently with the MapReduce framework as illustrated in Figure 2. The web session extraction can be performed independently at different nodes to achieve maximum parallelism.

Besides MapReduce, large-scale parallel databases can handle large datasets as well. However, the schema of our web session data changes frequently and is considered as semi-structured. Therefore, we chose MapReduce over databases for the benefit of flexibility in schema. The most popular open source implementation of MapReduce is Hadoop [15]. At eBay, we developed a data analytics platform called Mobius [11] on top of Hadoop. The Mobius

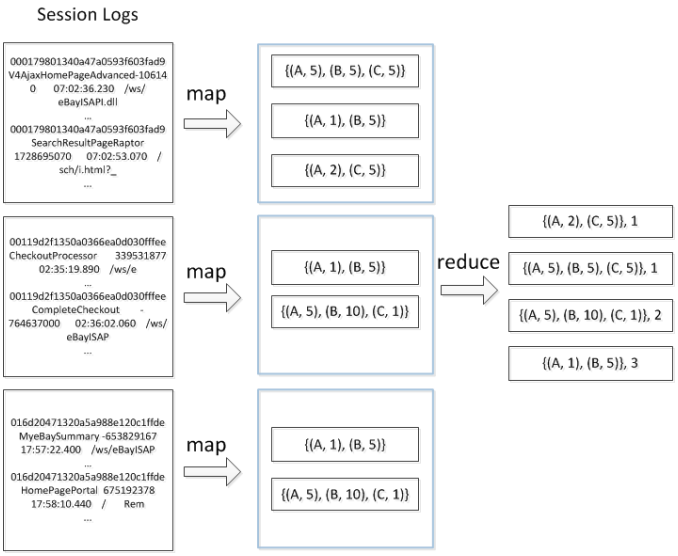

Figure 2: Data Filtering Using MapReduce. The web session data are split into chucks and processed in parallel on nodes across a cluster. The sessions satisfying the user-defined conditions are extracted and composed in the map step. $\{(A,1),(B,5)\}$ denotes a session, in which event A followed by event B. The elapsed time for A is 1 second, while that for B is 5 seconds. At the reduce step, the occurrence of each unique web session is counted by combining the sessions from the map step. The extracted web sessions with the number of occurrences as the final results of the query room.

platform incorporates powerful pattern matching operators that can efficiently find sub-sessions that matches the user-defined patterns[1]. The Mobius platform runs on a 1000-node shared Hadoop cluster. In practice, we run analysis on one-day web session data, which is about 6 terabytes large and contains over 900 million sessions. The query can return the results in about 4 minutes, which is acceptable for the analysts who are using our system. Please note that our visual analytics system is independent from the Mobius platform, which can be replaced with any large-scale data processing platform, as long as it meets the scalability requirement. The extracted data of interest is often in the magnitude of hundreds of megabytes, which can fit into the memory of a personal computer for interactive visualization.

4.2 User Interface

The visual interface of *TrailExplorer2* contains four components, i.e., the main view, the detail information panel, the legend and the distribution chart of the elapsed time, as shown in Figure 3. The user sessions are visualized in the main view. Besides the detail view of individual sessions, aggregated statistics are shown in the detail information panel and the distribution chart. The legend view shows the different colors used to represent all the possibly occurring events.

4.2.1 Session Data Visualization

The session visualization mainly aims to deliver the information of the ordering and elapsed time of the event sequences. We adopt a stacked bar chart representation, in which a session is represented as a horizontal stacked bar corresponding to the sequence of events. The length of each bar indicates the elapsed time of the corresponding event.

[1]The details of the Mobius platform are out of the scope of this paper. Please refer to [11] for more information.

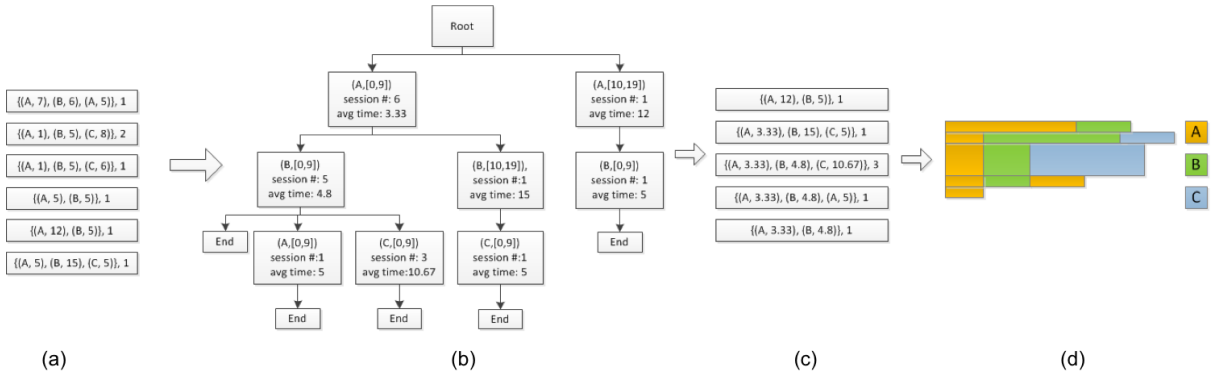

Figure 4: Building a tree structure to accommodate the input web sessions. (a) Input web sessions. (b) Starting from the first step of each session, each event is placed at one tree node. Suppose an event is at the i^{th} position away from the reference alignment point, it will be assigned to the $(i+1)^{th}$ level. Each tree node has an event type and a time bucket to decide whether an event can be placed within it. The last assigned event will be at a leaf node. In the process of tree building, we record the number of events stored at one node and the average elapsed time. (c) The aggregated web sessions. (d) Visualizing the aggregated sessions as horizontal stacked bars. The height of the bar indicates the number of sessions within the group, and the width indicates the elapsed time. Different events are colored differently.

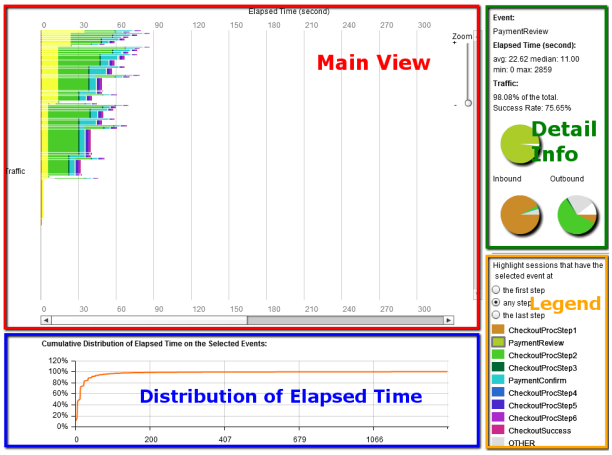

Figure 3: The user interface includes four components: the main view, the detail information panel, the legend and the distribution chart of the elapsed time.

Considering the sheer amount of web sessions, we aggregate and classify the data at different levels to enable a level-of-detail visualization. To this end, we employ a tree data structure to accommodate the input web sessions. Suppose we have a set of web sessions, as shown in Figure 4(a), where $\{(A,1),(B,5),(C,8),2\}$ means the event sequences $\{(A,1),(B,5),(C,8)\}$ occurred twice in the data. $\{(A,1),(B,5),(C,8)\}$ represents a sequence of events A, B, and C, taking time 1, 5, and 8 seconds respectively. We treat a session as a string and the event-and-time pair, e.g., $(A,1)$ as a character. Thus, a tree structure like prefix trees can be built as illustrated in Figure 4(b). The difference is that each node is associated with an event type and a time bucket, e.g., $(A,[0,9])$, instead of a scalar value. An event is assigned to a tree node if their event types match and the elapsed event time is within the node's time bucket. A depth first traverse of the tree creates sorted session groups that satisfy the requirement for visualization (See Figure 4(c)). In the visualization stage, each session group is drawn as a horizontal stacked bar. The height of the bar indicates the number of sessions within the group, and the width indicates the elapsed time. Different events are colored differently (See Figure 4(d)).

4.2.2 User Interaction

Interaction is the key in a visual analysis environment. Our system provides a wide range of interaction tools to facilitate thorough data exploration.

First, analysts can highlight the events of interest to investigate their characteristics. The highlighted events are colored more brightly. The detail information panel and the elapsed time distribution chart show the selected events' names, the average, median and cumulative distribution of their elapsed time. The total count and success rate of the sessions that contains them are displayed as well. Therefore, analysts can learn the correlations between the elapsed time and the success rate. The distributions of the inbound and outbound pages of the highlighted events are also illustrated as pie charts, which provide analysts insights of user behavior patterns. Analysts can highlight events through two different interactions. However, the system only allows them to highlight the same type of events at a time. They can select an event type in the legend. Based on the highlighting configuration, the sessions that have the selected event as a particular step are highlighted. For example, in Figure 3, all the PaymentReview events are highlighted with a brighter yellow color by selecting the event in the legend. In addition, analysts can also highlight events by selecting the stacked bars in the main view. In Figure 7(a), the group of PaymentReview events with elapsed time of 13 seconds is highlighted.

Second, using the default prefix tree, the output sessions are aligned upon their first events. The system allows analysts to select any event as an alignment reference to adjust the visualization layout. Only those sessions that contain the selected event will be displayed. In addition, the grouping of the sessions changes as the alignment changes (See Figure 5). This interaction allows analysts to investigate user behavior patterns conditioning on the characteristics of the event selected for alignment. In order to achieve this, the underlying tree has to be rebuilt for the new alignment configuration. We perform a circular shift for all the sessions to move the focus events to the first position. We then rebuild the tree using the shifted sessions. At the visualization stage, we will shift the sessions back to the original positions before drawing them. With all the data in memory, rebuilding the tree takes about 1.5 seconds on average in our experiments, which satisfies the interactive requirement.

Another interaction is modifying the time bucket size, which enables analysts to visualize and investigate data with different levels of granularity. The analysts can start with a rather coarse overview,

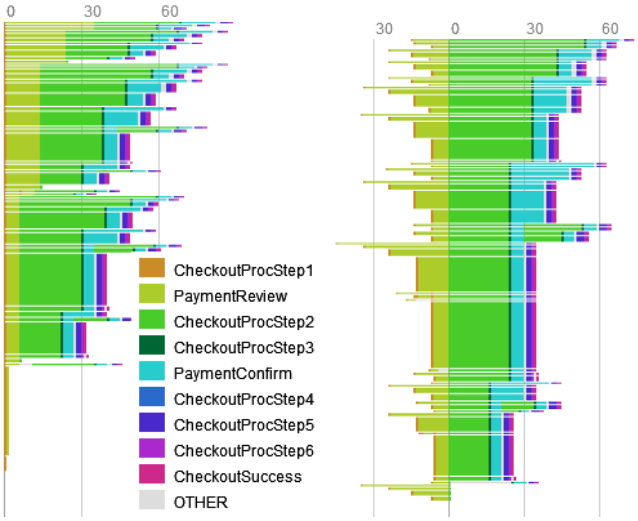

Figure 5: Comparing the visualizations of the same session data with different alignments. 5(a) and 5(b) show the results of aligning sessions using the CheckoutProcStep1 event and CheckoutProcStep2 event respectively.

generated with a large time bucket size. If they would like to see more granular distribution of the elapsed time, they can reduce the time bucket size and see more details. Usually, the tree needs to be rebuilt with the new time bucket size. Same as the rebuilding for alignment, it can be completed in real-time with less than 2 seconds latency. If the new time bucket size is a multiple of the original bucket size, the new session groups can be generated by merging the subtrees instead of rebuilding. This action can be completed in milliseconds.

5 Case Study

Our work was motivated by the fact that analysts have difficulties in exploring the vast amount of web session data to obtain insights of user behavior patterns. We invited some of them to use *TrailExplorer2* on the data of their interests, and collected feedback. In these studies, we were interested in the following questions:

- Does interactive visual exploration of the entire data set help analysts obtain better insights?
- How much do they like the experience of different response time for different user interactions?
- Are the visualization and interaction intuitive and useful?

In this section, we present our observations of how our system was used in two case studies [2].

5.1 Correlations Between the Elapsed Time and the Success Rate in the Checkout Flow

As illustrated in Figure 6, after committing to a purchase, a user is directed to the checkout flow. A webpage *flow* is defined as a sequence pattern, which contains a set of events and ordering dependencies among them. The events of a flow are also called steps. The checkout flow is one of the most critical flows at eBay. It starts

[2]Although real data were used in the studies, the results presented in this section are generated from a data sample that preserves the patterns in the original data. They are for illustration purpose and do not reflect the real performance of eBay services.

Figure 6: The checkout flow has three main steps in the flow, i.e., PaymentReview, PaymentConfirm and CheckoutSuccess. There are several intermediate events between these steps.

from the PaymentReview page, which allows the user to revise the payment amount and method. In CheckoutProcStep2 and CheckoutProcStep3, the user is asked to fill detail payment information of the selected payment method. Then, he/she needs to confirm the payment at the PaymentConfirm step. If the user finally reaches the CheckoutSuccess step, it is considered to be a successful checkout. The success rate of the checkout flow directly impacts eBay's revenue. Therefore, the checkout team would like to understand the factors that could potentially affect the success rate. They were particularly interested in the elapsed time of each event and its correlation with the success rate.

We introduced *TrailExplorer2* to a group of user experience designers and researchers from the checkout team. After an interactive demo session, they quickly learned the tool and started right away by selecting a set of checkout pages. They waited several minutes for the data filtering process to finish. They found the visualization quite intuitive and gave them a nice overview of the temporal user behavior patterns in the checkout flow. Then, they investigated different checkout steps by highlighting them on the visualization. Finally, they found that the PaymentReview step had the closest correlation with the success rate among all the steps. The longer the elapsed time at the PaymentReview step was, the less likely users tended to complete the checkout flow successfully (See Figure 7). Based on this insight, the team decided to simplify the PaymentReview page to reduce the elapsed time on the page. This was the first attempt by the team to use a visual analytics tool for understanding user experience. They were satisfied with the results and saw great potential of the tool.

5.2 Category Selection in the Listing Process

For eBay marketplace, listing items for sale is the beginning of all seller activities. During the listing process, users may visit following pages: *SearchCategory*, find the right eBay category for the item through keyword search; *BrowseCategory*, find the right category through browsing eBay's category hierarchy; *CatalogSearch*, find products that matches the item by searching eBay's product catalog; *CatalogSearchResults*, select one of the matching products from catalog search; *DescribeItem*, write description about the item and fill in selling related information, e.g., price, shipping, and etc; *ReviewEnhance*, review the listing and submit it; *Congratulation*: confirm that the listing is created successfully. The success rate of the listing process is defined as the percentage of users who reach the Congratulation page. Unlike the checkout, the listing process is not a structured webpage flow. Some of the pages listed above are optional. Users can take different paths to create the listing.

Making the listing process as frictionless as possible for users and improving the success rate are the top priorities for the selling team. We invited an analyst to try out *TrailExplorer2*. He had extensive experience with analyzing the listing process through querying databases. He had found that the DescribeItem page was the most time consuming step. Users spent quite some time on selecting the categories as well, and the elapsed time was negatively correlated with the success rate. He hoped to obtain more insights using our system. Based on the page list above, the system created the visualization as shown in Figure 8. The analyst started with verifying his previous findings in our system. He simply selected each page in the legend and was able to see not only the average elapsed time

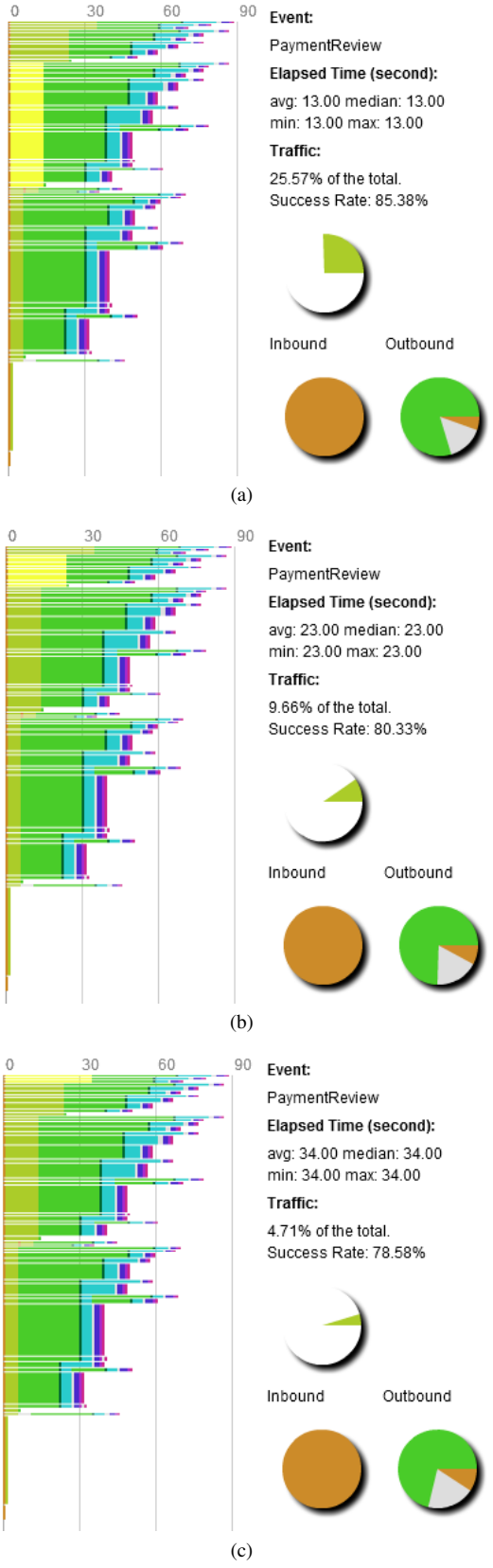

Figure 7: Correlations between the elapsed time of the PaymentReview step and the success rate. In 7(a), the average elapsed time of the highlighted PaymentReview events is 13 seconds as shown in the detail information panel. It also shows that the success rate of these sessions is 85.38%. In 7(b) and 7(c), as the elapsed time of the same step increases to 23 and 34 seconds, the success rate drops to 80.33% and 78.58%, respectively.

of each page in the detail information panel, but also its cumulative distribution in the distribution chart. He then highlighted groups of sessions with various elapsed time of a particular page to check the corresponding success rates as illustrated in Figure 8 [3]. Within several mouse clicks, he quickly reached the same conclusions as those from his previous studies.

As the first and a required step of the listing process, category selection is critical. Moreover, it can be challenging to novice sellers due to the massive category hierarchy. As mentioned before, there are different ways to finding a proper category: searching (i.e., SearchCategory), browsing (i.e., BrowseCategory) and selecting a product in the catalog (i.e., CatalogSearch and CatalogSearchResults). In this step, sellers can go through any of those three ways and switch between them as they want. The analyst would like to understand the different paths sellers took in category selection and the impacts on their listing experience.

He aligned the sessions by the BrowseCategory page to investigate the behavior patterns of sellers who chose to find category through browsing. Several interesting user behavior patterns, which were invisible before, emerged. In Figure 9, we annotate some of them using color boxes. For example, the users in the red box, which was almost half of all the sellers browsed, started with browsing directly and took less than 20 seconds on average to finish the listing process. They found the categories through browsing without searching and filled the descriptions in several seconds. The analyst believed that these were very experienced sellers, who knew which category they wanted and simply pasted their saved item descriptions. The blue box circles a group of users dropped off after the BrowseCategory page. Comparing to the others, they spent much longer time on average on the SearchCategory page before browsing. The analyst thought that it suggested that browsing could not help sellers when they already experienced too many difficulties with category searching.

He also aligned the sessions by the DescribeItem page (See Figure 10). The sessions were sorted by the elapsed time of the DescribeItem page in descending order. We annotate the interesting part using a red box in the figure. The analyst found that for the sellers who spent less time on the DescribeItem page, more of them found the category through the catalog searching (i.e., CatalogSearch and CatalogSearchResults). In addition, sellers who spent more time on the DescribeItem page also spent more time on the SearchCategory page. It suggested that finding the matching product in the catalog helps making the listing process easier. The analyst was very happy with these insights. He really enjoyed the visual exploration experience. The system allowed the analysts to easily uncover valuable patterns which he could not find using the iterative query-based analysis before.

5.3 Feedback

The participants in two studies were quite different. The participants in the first study were not experts in web analytics. They were really happy with the system because it enabled them to conduct deep dive analysis without writing any analysis codes or queries, which they were not good at. However, the system was also valuable to the expert from the selling team in the second study. He learned some new insights from the visual exploration as well.

As discussed in Section 3.3, we have to make a compromise between interactivity and scalability. We choose to perform the data filtering on a cluster and expect the response latency in minutes. Its impact on the system's usability is our major concern, since users often expect real-time interaction when they use a GUI system. In our studies, the first group was a little bit confused until further explanation. For the expert user who had dealt with the large-scale

[3]Due to the limited space in the paper, we choose to annotate the highlighted events with the corresponding detail information panels on the figure.

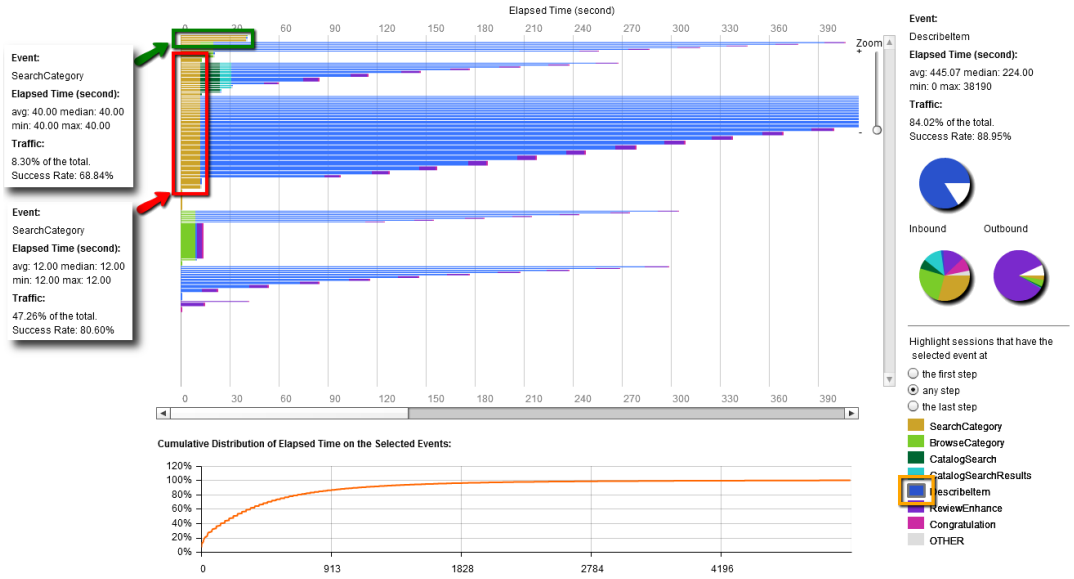

Figure 8: Visualization of the user temporal behaviors in the selling flow. The DescribeItem page (i.e., the blue bars) is highlighted. It takes users the longest time among all the pages. The distribution chart at the bottom shows that about 50% users spend more than 4 minutes. The analysts also highlighted the SearchCategory events in the red box and blue box separately, and checked the corresponding statistics from the detail information panel. The detail panels on the left show that the longer users spend on the SearchCategory page, the more likely they drop off.

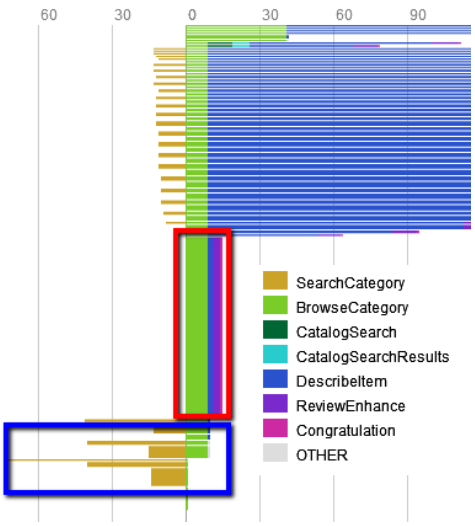

Figure 9: Visualization of the sessions aligned by the BrowseCategory page. The sellers in the red box start the listing with browsing and successfully complete it in a short time. On the contrary, those in the blue box spend a longer time on average in searching before switch to browsing, and drop off afterwards.

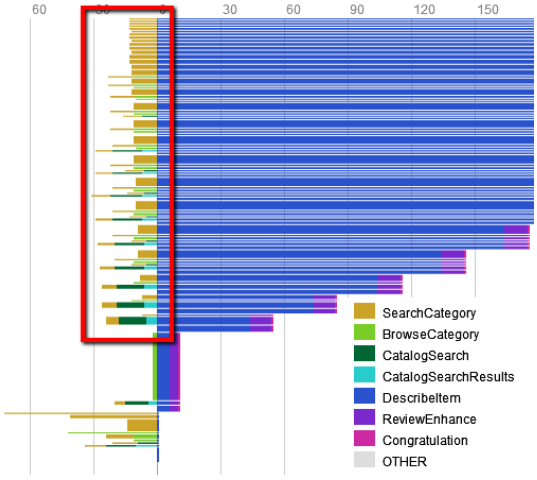

Figure 10: Visualization of the sessions aligned by the DescribeItem page. As the elapsed time of the DescribeItem page decreases from top down, the length and width of the yellow (SearchCategory) and light green (BrowseCategory) bars decrease, while those of the dark green (CatalogSearch) and light blue (CatalogSearchResults) bars increase. It suggests the negative correlation between the elapsed time of the DescribeItem page and the usage of the catalog search.

data, there was no problem at all. He was used to wait for 10 minutes or longer for his queries to complete. However, he suggested make the data filtering operation non-blocking. Users can submit a list of selected pages and be notified later through email when the filtering is finished. Then, they can return to retrieve the extracted user sessions for analysis.

All the participants found the visualization to be intuitive. "It is just like the Gantt chart we use for project management" one participant said. The interactions were proven to be useful too. As we shown in the case studies, the alignment functionality was very powerful in investigating behavior patterns conditioning on user behavior on a certain page. The data filtering and time bucket adjusting were useful for users to find the best view for their data. For example, the checkout team selected all the checkout pages at the beginning. Through the study, they identified couple trivial pages and removed them by filtering. It greatly reduced the visual complexity of the visualization. They also had to adjust the time bucket size to find the proper granularity, because they had little idea about the elapsed time in the checkout flow. These interactions helped them to obtain the optimal view through an exploratory analysis process.

6 Conclusions and Future Work

In this paper, we introduce a two-tier visual analytics system, which incorporates a Hadoop-based parallel data processing platform to handle queries of massive web session data. We also discuss the process and principles we follow in designing the system. We observe how analysts conduct analysis on such data, and identify the major analytics tasks. For these tasks, we analyze the amount of data they access, and implement them at different tiers. Finally, we demonstrate the usefulness of the system through real-world case studies. We believe not only this kind of two-tier approach but also the designing process can be generalized to deal with other large data visual analysis problems.

During our user studies and experiments, we found that our visualization could not represent certain behavior patterns very well, e.g., a long session with a large number of recurring patterns. A more clever visual representation is desired to illustrate them in a compact way. In particular, we would like to exploit hierarchal data structures to visually organize and display different types of patterns with various lengths. Moreover, a visual indication of the success rate in the main visualization could make the exploration more efficient. In future, we also plan to extend the current system to support more sophisticated analysis tasks by feeding user visual analytics results back to refine the Hadoop-based query and complete the iterative loop of data exploration.

Acknowledgements

Jishang Wei worked on this project as a summer intern at eBay Research Labs. This work has also been supported in part by the U.S. National Science Foundation through grants CCF-0811422, IIS-1147363, CCF-0808896, and CCF-1025269, and also by the U.S. Department of Energy through the SciDAC program with Agreement No. DE-FC02-06ER25777 and DE-FC02-12ER26072, program manager Lucy Nowell.

References

[1] N. V. Andrienko and G. L. Andrienko. Designing visual analytics methods for massive collections of movement data. *Cartographica*, 42(2):117–138, 2007.

[2] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[3] D. A. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler. Challenges in visual data analysis. In *Proceedings of the conference on Information Visualization*, IV '06, pages 9–16, Washington, DC, USA, 2006. IEEE Computer Society.

[4] M. Krstajic, E. Bertini, and D. Keim. Cloudlines: Compact display of event episodes in multiple time-series. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2432–2439, Dec. 2011.

[5] H. Lam, D. Russell, D. Tang, and T. Munzner. Session viewer: Visual exploratory analysis of web session logs. In *Proceedings of the 2007 IEEE Symposium on Visual Analytics Science and Technology*, VAST '07, pages 147–154, Washington, DC, USA, 2007. IEEE Computer Society.

[6] B. Lee, C. S. Parr, C. Plaisant, B. B. Bederson, V. D. Veksler, W. D. Gray, and C. Kotfila. Treeplus: Interactive exploration of networks with enhanced tree layouts. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1414–1426, Nov. 2006.

[7] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman. Lifelines: visualizing personal histories. In *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, CHI '96, pages 221–227, New York, NY, USA, 1996. ACM.

[8] M. Schaefer, F. Wanner, F. Mansmann, C. Scheible, V. Stennett, A. T. Hasselrot, and D. A. Keim. Visual Pattern Discovery in Timed Event Data. In *Proceedings of Conference on Visualization and Data Analysis*. SPIE, 2011.

[9] Z. Shen and N. Sundaresan. Trail explorer: Understanding user experience in webpage flows. In *IEEE VisWeek Discovery Exhibition*, Salt Lake City, Utah, USA, 2010.

[10] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, pages 336–343, Washington, DC, USA, 1996. IEEE Computer Society.

[11] N. Sundaresan, E. Chiu, and S. Gyanit. *Scalable stream processing and MapReduce*. Hadoop World NY, 2009.

[12] F. van Ham and A. Perer. Search, show context, expand on demand: Supporting large graph exploration with degree-of-interest. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):953–960, Nov. 2009.

[13] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman. Aligning temporal data by sentinel events: discovering patterns in electronic health records. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 457–466, New York, NY, USA, 2008. ACM.

[14] T. D. Wang, C. Plaisant, B. Shneiderman, N. Spring, D. Roseman, G. Marchand, V. Mukherjee, and M. Smith. Temporal summaries: Supporting temporal categorical searching, aggregation and comparison. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1049–1056, Nov. 2009.

[15] T. White. *Hadoop: The definitive guide*. Yahoo Press, 2010.

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

[17] J. wook Ahn, P. P. Brusilovsky, and K. Wongsuphasawat. Analyzing user behavior patterns in adaptive exploratory search systems with lifeflow. In *Proc. Workshop on Human-Computer Interaction and Information Retrieval (HCIR)*, 2011.

[18] J. Zhang and S. You. Dynamic tiled map services: supporting query-based visualization of large-scale raster geospatial data. In *Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research and Application*, COM.Geo '10, pages 19:1–19:8, New York, NY, USA, 2010. ACM.