# LoyalTracker: Visualizing Loyalty Dynamics in Search Engines

Conglei Shi, Yingcai Wu, *Member, IEEE*, Shixia Liu, *Senior Member, IEEE*, Hong Zhou and Huamin Qu, *Member, IEEE*

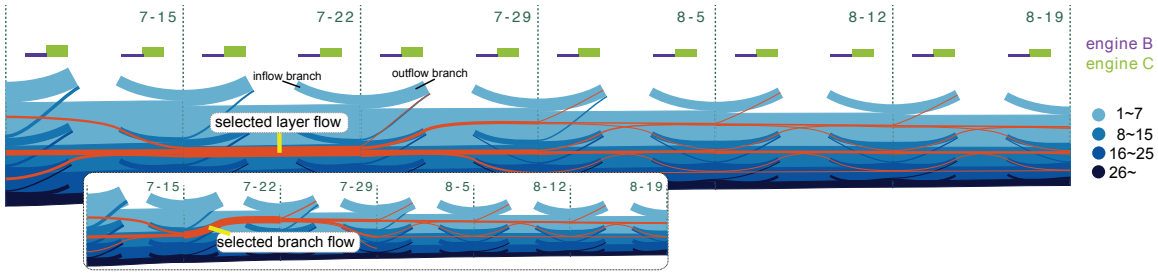


Fig. 1: LoyalTracker illustrates loyalty dynamics of the users using search engine **A**. Top and bottom show the same flow view that highlights two different flowing patterns of the users (in orange) selected from a layer flow (top) and a branch flow (bottom) across multiple loyalty categories (layers) over time. The switching histogram on the top shows a visual summary of switching behavior.

**Abstract**—The huge amount of user log data collected by search engine providers creates new opportunities to understand user loyalty and defection behavior at an unprecedented scale. However, this also poses a great challenge to analyze the behavior and glean insights into the complex, large data. In this paper, we introduce LoyalTracker, a visual analytics system to track user loyalty and switching behavior towards multiple search engines from the vast amount of user log data. We propose a new interactive visualization technique (flow view) based on a flow metaphor, which conveys a proper visual summary of the dynamics of user loyalty of thousands of users over time. Two other visualization techniques, a density map and a word cloud, are integrated to enable analysts to gain further insights into the patterns identified by the flow view. Case studies and the interview with domain experts are conducted to demonstrate the usefulness of our technique in understanding user loyalty and switching behavior in search engines.

**Index Terms**—Time-series visualization, stacked graphs, log data visualization, text visualization

## 1 Introduction

Search engines have become a necessity in our daily life, as the search engines empower us to find our desired information rapidly from the vast volume of web pages. Search engine business has thus become one of the most profitable businesses on Internet. Meanwhile, search engine providers also face fierce market competition [17, 27]. Prior studies show that 70% of users rely on multiple search engines and due to the low switching cost, engine switching happens frequently [44]. Earning loyalty can be critically important for search engine providers. Customer loyalty has long been regarded as a vital source of sustained profits, which enables a company to develop a sustainable advantage over competitors [1]. Retaining merely 5% of customers can improve profits by almost 100% [28]. Therefore, effectively tracking user loyalty and understanding how and why loyalty changes and users leave become particularly indispensable for search engine providers.

As online service providers, search engine companies can collect large-scale log data from their users who consent to provide their search data, offering a richer source of data for an in-depth analysis of user behavior. This capability attracts considerable research attention from different research areas such as data mining and visualization. Extensive studies have been conducted to analyze the user behavior with respects to search engine switching [9, 42, 43, 44]. However, most existing work mainly focuses on short-term search engine switching behavior [43] and uses simple statistical methods to validate some given assumptions rather than detect abnormal or unexpected patterns. Visualization practitioners have also developed various systems [26, 40, 21, 34] using visualization techniques such as graphs to explore user behavior from the collected user log data. Nevertheless, most systems aim at visualizing Web traffic and/or user navigation paths through a website, but are incapable of tracking user retention rigorously or conducting a systematic analysis of defection patterns.

A system capable of tracking customer loyalty is critically important for companies to prevent customer defection. For instance, customers usually have a higher probability of leaving when loyalty is found to be continuously decreasing. With an effective visualization system, a company can make an informed decision to customize its offerings or to take other necessary actions to retain customers. However, there are few proper visualization techniques that can be used to solve the problem. One particular challenge is the design of an intuitive and interactive visual representation to understand long-term behavior better, as characterized by dynamic loyalty variation and frequent defection over time [27, 43]. In addition, the faster word-of-mouth and lower switching costs on the Internet change the pace at which companies must improve their products and services to keep users loyal. These goals require timely detection and a thorough analysis of user loyalty and defection [27]. The growing scalability of the search log data makes timely detection and analysis difficult. Despite the discovery of patterns, the manner by which to convey the findings successfully to managers is another obstacle. Therefore, developing an effective and intuitive system to glean insights into large scale data through the visualization of user loyalty and defection over time is important.

To address these challenges, we develop LoyalTracker, a visual analytics system to analyze and track user loyalty and switching behavior towards multiple search engines using large-scale user log data. LoyalTracker enables analysts to define multiple loyalty categories such as hard-core loyals and switchers [18], following the general practice in marketing. The analysts can then visually trace how users change their loyalty across categories over time. The system has three linked

- *C. Shi and H. Qu are with the Hong Kong University of Science and Technology. E-mail: {clshi,huamin}@cse.ust.hk*
- *Y. Wu and S. Liu are with Microsoft Research Asia. E-mail: {yingcai.wu, shixia.liu}@microsoft.com. S. Liu is the corresponding author.*
- *H. Zhou is with Shenzhen University. E-mail: hzhou@szu.edu.cn*

*Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014. Date of publication 11 Aug. 2014; date of current version 9 Nov. 2014.*
*For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.*
*Digital Object Identifier 10.1109/TVCG.2014.2346912*

views: a flow view, a density map view, and a word cloud view. The flow view uses a new interactive visualization technique inspired by a chart drawn by Munroe [24], which can visualize the dynamics in each predefined loyalty category and their aggregation over time clearly. In the density map view and word cloud view, we show more detailed information on demand, which aids analysts in quickly detecting user behavior, determining the underlying reasons, and intuitively conveying the findings to a wide audience.

LoyalTracker targets the tracking and analysis of the user loyalty on search engines, but techniques used can be easily adapted to other similar problems on user loyalty and defection, because the characteristics of loyalty and switching behavior are shared across different areas.

Our contributions are described as follows:

- a set of design principles for analyzing the long-term and dynamic user loyalty of a large number of users at different levels of details.
- an interactive visualization system to enable analysts to understand better user loyalty deviations within a search engine and the defections between multiple search engines.
- an augmented stacked graph to show the flows between layers and the flows that enter or leave the graph.

## 2 Related Work

Past research related to this work can be classified as follows: search engine switching, stacked graphs, and log data visualization.

### 2.1 Search Engine Switching

Web search engines aid users in quickly locating information on the Internet, making the tools a necessity in daily life. A large percentage of daily users (up to 70%) frequently switch between different engines because of the low switching cost and other factors such as satisfaction, effectiveness, and familiarity [44]. In addition, 4% of search tasks involve multiple search engine usages. This percentage increases to over 10% for longer search tasks [42]. Extensive studies were conducted to understand how, when, and why users switch engines [9, 42, 43, 44]. Prior studies classified users into different categories based on their switching behavior [16], examined multiple search usage [44], developed models to explain engine switching behavior based on brand loyalty [23], predicted when users will switch [9], or employed large-scale log analysis and user survey data to understand user motivations for switching [42]. Researchers have recently suggested the development of a better understanding of long-term engine switching behavior [43], but existing studies only use simple statistical methods to validate the given assumptions. To our knowledge, interactive visualization systems capable of tracking and analyzing user behavior based on loyalty and defection remain lacking.

### 2.2 Stacked Graphs

A stacked graph is constructed by stacking one time series on top of another, such that each time series is represented by a stacked layer. Havre et al. [13] introduced ThemeRiver that uses a river metaphor to visually depict the thematic variations in documents over time. Byron and Wattenberg [2] presented Streamgraph, which significantly improves the quality of standard stacked graphs. Streamgraph has been extended to help analysts better understand text corpora [5, 6, 22, 31]. TIARA [22] and its variant [31] are interactive visual text analysis tools that seamlessly integrate text summarization techniques for exploring large collections of documents. Dörk et al. [6] described a highly interactive system based on tailored stacked graphs to visualize a continuously updating information stream. These techniques can be used to visualize multiple time series but cannot reveal how different time series exchange quantities over time.

TextFlow [5] employs a flow-based metaphor to enhance stacked graphs, such that the merging and splitting relationships between evolving topics in text corpora can be revealed. RankExplorer [30] extends stacked graphs by embedding color bars and changing glyphs to help users analyze ranking changes. TextFlow and RankExplorer were developed for different applications, but both could be used to track how quantities of time series flow to one another using either the flow-based visual metaphor [5] or the color bars [30]. However, the two methods are inappropriate for tracking user loyalty because of a variety of limitations such as distortion and intuitiveness issues. We analyze and compare the techniques in more detail in Section 5.1.1.

### 2.3 Log Data Visualization

Human-generated log data, such as query logs [48], usability logs [10, 8], and Web clickstreams [41], records various activities of users. SDSS Log Viewer [48] helps analysts quickly identify the data-seeking behavior of users from SQL query logs. Gray et al. [8] used a set of vertical striped bars to show the usage patterns from usability logs collected from a graphical user interface. Clickstreams visualization examines Web traffic and/or user navigation paths through a Web site for user behavior analyses [3, 20] and usability improvement [4]. Numerous systems such as WebQuilt [40] and Webviz [26] commonly use trees, treemaps, or node-link graphs to visualize clickstreams. Lee et al. [20] used parallel coordinates and star fields to visualize user paths and product performance. Wei et al. [41] introduced an interactive clustering method to reveal user behavior patterns from Web clickstream data. TrailExplorer2 [29] uses stacked bars and pie charts to discover valuable information from large-scale Web clickstreams.

Behavior Graphs (WBG) [3], based on state diagrams, were used to visualize the search structure on the Web and to discover usage behavior patterns. Lam et al. [19] developed Session Viewer to help analysts understand Web search usage behavior with multiple coordinated views, such as state transition diagrams, stacked bars, and tables. Outflow visualization [45] used a flow based design to show the aggregated multiple event sequences and their outcome, which can be used to analyze event progression pathways. Masruri et al. [36] improved the directed graphs to summarize people's preference transition. The improved graphs can show more information like latency and frequency, but it can hardly show long time temporal patterns. Compared with existing systems, we mainly focus on analyzing the dynamics of user loyalty and defection using millions of Web log entries, which is not supported by previous systems.

## 3 Background

This section introduces the background knowledge on user loyalty and search engine switching, and then summarizes the analysis tasks.

### 3.1 Customer Loyalty and Defection

A company can significantly boost its profits by building customer loyalty and reducing customer defection [28]. Thus, companies need to gather information about customers to track their loyalty, analyze their behaviors, and identify why they are leaving [28]. Customer loyalty is a complex phenomenon and is therefore difficult to define and measure [46]. From a behavioral view, customers are considered loyal to a firm if they consistently purchase products or services from the firm. ***Behavioral measures*** contain criteria such as repeat purchase and word-of-mouth referrals. From an attitudinal view, customers are considered loyal to a firm if they have a strong desire to maintain a relationship with the firm. ***Attitudinal measures*** include criteria such as commitment and trust. Attitudinal measures may better explain how and why loyalty changes but are more difficult to evaluate quantitatively and may need additional efforts to conduct questionnaires.

Loyalty analysis of search engines has received considerable attention [9]. Search engines have a significantly lower switching barrier, and users can easily change their search engines [9, 43]. Loyalty change and switching behavior are also easily observed, thus providing relatively complete data. Frequent variations and the rich data create a good opportunity to analyze the loyalty and derive design principles. Therefore, we base our work on the loyalty analysis of search engines. Nevertheless, our visualization design can be easily extended to analyze loyalty and switching behavior of other products and services, because of similar data characteristics, analysis tasks, and goals.

### 3.2 Data Characterization and Task Analysis

We collected search log data from consenting users of a widely-distributed Web browser and store it in a MapReduce system. All personally identifiable information from the logs had been removed.

Every log entry includes: a unique user identifier for each user, a query performed by the user, the search engine used for the query, the timestamp when the query was issued, and the URL and dwell time of the result page clicked.

We followed an iterative user-centered design process to develop our visualization techniques. We worked closely with three domain experts (two applied scientists (AS) and a software development engineer (SDE)), from the search department of a corporation for eight months. The ASs focus on analyzing and understanding online user behavior and user experience. The SDE maintains a system called *XSystem*. *XSystem* uses the 2.5% sampled data from the collected raw data. It enables analysts to access large-scale search log data quickly using carefully-constructed queries and returns the results in a table.

The feedback collected by the SDE suggests that a visual system that enables analysts to interact with the complete data and explore searching behavior is urgently needed. We held biweekly meetings and exchanged emails with the experts to gather and refine design requirements, present prototypes, and collect feedback to improve the system iteratively. We define some terms formally below.

- A *search session* consists of a series of user search activities. A session ends if the user is idle for more than 30 minutes, which is widely adopted [33, 42].
- A *switching event* is defined as a pair of consecutive queries issued on distinct search engines within a single search session [9].

According to the suggestions of the domain experts, we employ well-established measures to estimate user loyalty and satisfaction.

- *User loyalty in search* is evaluated using a behavioral measure of user engagement, the frequency of using the search engine. In general, there are two widely used metrics to quantify the user engagement: the number of queries [14] (at query level) and the number of sessions [33] (at session level) performed by users in certain time periods. The user engagement at query level is mainly used to describe the short-term user behavior, whereas the user engagement at session level is focusing more on describing the long-term user behavior. In our case, the analysts pay more attention to long-term behavior analysis. Therefore, the loyalty is defined at session level.
- *User satisfaction in search* is measured based on how long a user stays on the destination page. A query is viewed as satisfactory if the searcher clicks a search result followed by a dwell time of more than 30 seconds [7, 11, 15, 42]. Although Hassan et al. proposed a sophisticated metric to measure user satisfaction [12], we chose the simpler one because of two concerns: 1) due to the large size of the data and the computing complexity of the sophisticated metric, in the current computing infrastructure, it is hard to get the result in reasonable time when applying the sophisticated metric; 2) the simpler metric is widely used in the company and it has been proved to be effective. Nevertheless, in our system, we have carefully decoupled the calculation module with other modules such that it will be easy to replace current measure with more advanced one without affecting other modules.

We complied a list of analysis tasks through a series of interviews with them. This process aided us in better understanding the problem domain and identifying the challenges faced by the target users.

Q.1 *How does the loyalty of searchers using a particular search engine change over time?* Groups of users with a similar loyalty-changing trend are of particular interest for analysis. Detecting not only the sudden and dramatic changes in user loyalty, but also the long-term and gradual loyalty changes is important.

Q.2 *How do searchers using a particular search engine switch to other search engines over time? How is the user-switching behavior related to the dynamic variation of user loyalty over time?* The analysis and identification of the user-switching behavior pattern is crucial for formulating effective strategies to retain users.

Q.3 *Where do the new users come from and have these users used the search engine before? How does the loyalty of switchers or new users change over time?* Our collaborators need to analyze the distribution of switchers or new users to gain a better understanding of user behavior.

Q.4 *What are the differences in the dynamic loyalty variation or user-switching behavior among multiple search engines?* Our collaborators want to analyze the differences among multiple search engines to identify the strengths and weaknesses of a search engine better based on loyalty and defection.

Q.5 *What are the reasons for the switching behavior or loyalty change?* Our collaborators want to determine the reasons behind the identified user behavior pattern. These experts are interested in knowing whether a change in user satisfaction could lead to the behavior or if any keywords that easily trigger the pattern exist.

## 4 System Design

In this section, we discuss the visualization challenges and design rationale for the system.

### 4.1 Analysis Challenges and Design Rationale

During our collaboration with the experts, we identified a few analysis challenges. Before we proposed the LoyalTracker system, when analysts want to analyze the complete data, they have to formulate their assumptions and construct scripts to test these assumptions and then submit the scripts to a MapReduce system to identify user behavior patterns. Running scripts on the system may take hours or days depending on the complexity of the tasks and the resources available on the system before the analysts can retrieve the results in tables. The analysts often use Microsoft Excel to analyze the retrieved table data and to create simple charts for demonstration. Obtaining the desired results to validate their assumptions is a tedious trial-and-error procedure. Testing assumptions is difficult, but detecting unexpected user behavior patterns from a vast amount of data is significantly more difficult and time-consuming. Although interesting results may be detected, effectively presenting these results would be another obstacle.

A visualization system that enables analysts to analyze user loyalty interactively is urgently needed by the analysts in the company. Existing visualization techniques are unsuitable for the effective tracking of dynamic loyalty variation and analyzing user-switching behavior in one coherent view. We worked closely with domain experts and identified a set of design goals to address these challenges.

A. **User Flow Revelation.** In marketing and loyalty analysis, analysts often classify users into different loyalty categories, such as hard-core loyals [18]. The users behavior is then analyzed and compared across different loyalty categories. The domain experts are concerned with the visual tracking of the dynamic variations of different loyalty categories and the summation of all the categories (Q1). More importantly, the design could effectively reveal how users flow between different loyalty categories as well as how they flow in and out of a typical search engine over time, thus enabling the accomplishment of the analysis tasks (Q1-Q3) related to the user flow of loyalty variation.

B. **Intuitive Storytelling Metaphor.** A visual metaphor capable of telling a story intuitively is desired by our collaborators for the analysis tasks (Q1-Q5). An appropriate visual storytelling metaphor enables them to convey their findings more effectively with the support of visual evidence or related details to product teams and senior managers. Therefore, our work employs a visual representation based on an intuitive flow metaphor to facilitate storytelling.

C. **Multi-Scale Visual Representation.** Detecting both short-term and long-term patterns is important. A sudden and dramatic change in customer loyalty should be immediately detected, especially for online service providers with a significantly lower switching barrier, to take prompt actions. It is also crucial for companies to identify long-term user behavior patterns, which can shed more light on user preferences and usage patterns [43]. Knowledge of the key trends in user loyalty and switching behavior is invaluable to companies. Therefore, the design should naturally support multi-scale analyses (Q1-Q5).

D. **Interactive Pattern Unfolding.** A visual system that enables analysts to interact with the data directly and see the results immediately is always preferred by domain experts to complete the described tasks, particularly for Q3-Q5. The system should provide a visual overview of how user loyalty changes over time to identify interesting patterns, and enable analysts to gain further insight into the patterns and determine

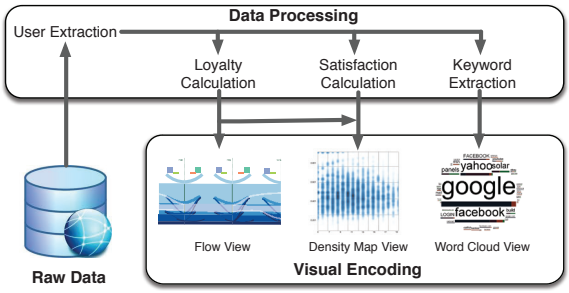


Fig. 2: Our system has a data processing part and a visualization part with three coordinated views. A flow view is used to provide a quick overview of the dynamic loyalty variation and switching behavior. A word cloud view and a density map view can help analysts interpret the user behavior patterns discovered in the flow view.

the causes of such changes (i.e., the relationships between satisfaction and loyalty, and keywords that trigger the patterns).

E. **Comparative Analysis.** A system that enables comparative analysis can aid in accomplishing the task of Q4. A desirable system should naturally support interactive brushing and visually highlight the similarities and differences in the dynamic user loyalty flow among different search engines. Furthermore, a statistical graph showing the overall distribution of the loyalty variation and flow in a search engine is also needed to facilitate the comparative analysis.

### 4.2 System Overview

LoyalTracker begins with the data processing part, which submits a script to the MapReduce system and retrieves the results once the tasks complete. After that, LoyalTracker can perform interactively. LoyalTracker consists of three views: a flow view, a density map view, and a word cloud view (Fig. 2). The flow view is used to provide a visual summary of dynamic loyalty change and user-switching pattern over time, such that analysts can track the dynamic variation readily. It uses a compact flow-based visual metaphor to enhance the stacked graphs for intuitively showing not only how users flow between layers but also how users enter or exit the search engine over time. The analysts can flexibly and visually specify queries in the flow view to perform an in-depth analysis in other views. The density map view can help analyze the relationships between user loyalty and satisfaction of a group of users. The word cloud view visually summarizes the query keywords that trigger engine switching for a group of users selected in the flow view to facilitate further understanding of the correlation between keywords and flow variation. The three views are well-coordinated to help analysts accomplish the various of tasks described in Section 3.2.

## 5 Visual Encoding Methods

In this section, we describe a set of visualization techniques for analyzing user loyalty. User interactions are subsequently presented.

### 5.1 Flow View

In the flow view, we propose a new design and an interaction technique based on a flow metaphor for visualizing the dynamics of user loyalty. The flow metaphor is inspired by an infographics chart drawn by an artist [24] to illustrate the history of the United States congressional elections. Fig. 3(a) shows the chart placed horizontally to fit the wide screen of modern displays better, with the timeline starting from left to right. The Congress members are categorized into different political groups, such as "left" and "right", based on their political beliefs. The chart visually traces the evolving composition of the US congress using a stacked graph layout. Each group is encoded using a layer in a distinct color. The graph, compared with general stacked graphs, also shows how different layers exchange members with one another using a flow metaphor (i.e., branches) over time. The chart shows two types of branches on top of a layer at each time point.

* **Inflow branch** along the top-left to bottom-right direction (Fig. 3(a)) indicates that a group of members is entering the layer from the above. The inflow coming from the top empty space (Fig. 3(b)) represents the new members.
* **Outflow branch** along the bottom-left to top-right direction (Fig. 3(c)) indicates that a group of members are leaving the layer to go to the layers above. The outflow departing from a layer to the top empty space (Fig. 3(d)) represents the leaving members.

An outflow branch linked to an inflow branch (Fig. 3(e)) indicates the returning members who left for a short time. The width of a branch represents the number of members in it. The endpoints of multiple inflow (or outflow) branches are bundled together if the branches meet in a layer during a time frame.

The chart is excellent for storytelling and has facilitated numerous discussions on the Web (approximately 430,000 results returned from the Google by searching "xkcd 1127" and "xkcd congress" as of March 2014). The stable political beliefs of members yield the chart with less edge crossings, resulting in a clear, legible layout.

The tasks of loyalty analysis are very similar to the tracing of political beliefs of the US congress members. A loyalty analysis also defines multiple loyalty categories and traces the dynamic change in different categories as discussed in Section 4.1. The customer loyalty does not dramatically change in most cases on the Web or in the real world [27]. The storytelling characteristic of the metaphor and its capability to reveal the flow between multiple categories directly satisfy design rationalities *A* and *B*. Therefore, we use a similar visual metaphor to design the flow view. We also extend the basic layout to enable multi-level representation, interactive pattern unfolding, and comparative analysis (meeting design rationality *C*, *D*, and *E*).

#### 5.1.1 Comparison with Alternative Solutions

The domain experts need to visually track the dynamics of different loyalty categories as well as the summation of all the categories (Q1). According to the design rationals of user flow revelation *(A)*, it is reasonable to design a visualization similar to stacked graphs to achieve the goal.

It also demands an advanced layout that can convey how users flow across multiple loyalty categories over time. Two existing stacked graph techniques, TextFlow [5] and RankExplorer [30], can accomplish this goal. TextFlow also uses a flow-based metaphor to convey the flowing pattern (see Fig. 4(a)). This technique draws a complete flow branch between two consecutive time points to represent the streams of customers, which might introduce significant visual clutter caused by the edge crossings of multiple flows. TextFlow optimizes the layer ordering to reduce visual clutter. However, the ordering in loyalty analysis should be preserved because the order inherently implies a semantic meaning (i.e., the loyalty level). By contrast, LoyalTracker only draws a partial flow rather than a complete flow to mitigate the problem of edge crossings. TextFlow produces irregular white gaps between layers to show the flow, which distort the layers and present an incorrect aggregate pattern (sum of individual time series). Our flow-based visualization does not produce white gaps.

RankExplorer [30] uses color bars rather than the flow metaphor to convey the flowing pattern (see Fig. 4(b)). A flow from a layer $m$ to another layer $n$ is represented by a color bar in $m$, wherein the height of the bar encodes the flow size and the color of the bar encodes the flow direction. Compared with TextFlow, the color bars can eliminate the problem of edge crossings and help create a more compact layout without the distortion. However, this method is generally less intuitive than the flow-based methods.

We present all three candidate designs in Fig. 4 to our collaborators and the feedback from them also confirmed our design choice.

#### 5.1.2 Visual Encoding

We use the flow metaphor described in Section 5.1 to design the flow view to show the dynamic loyalty variation. Fig. 4(c) shows the visual encoding scheme. We define a ***cell*** as the part of a layer in a time frame (i.e., from time point $t$ to $t+1$).

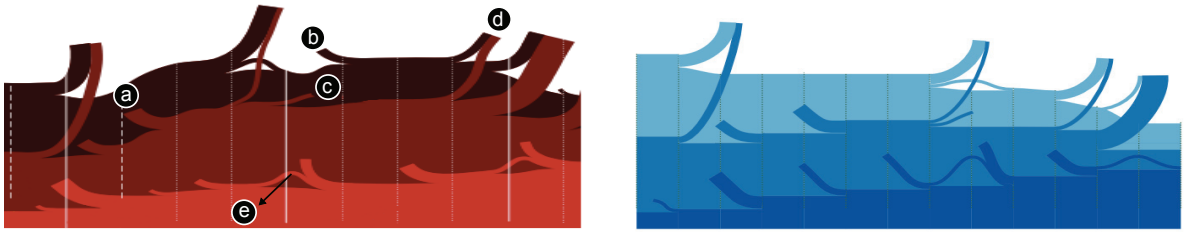


Fig. 3: Left: Layout extracted from a chart drawn by an artist [24] where (a) and (b) are the inflow branches; (c) and (d) are the outflow branches; and (e) is a group of returning members. Right: Layout created through our method using the data manually extracted from the infographics chart. Both layouts can show not only the dynamic variation of each group but also the dynamic exchange of members across different groups.

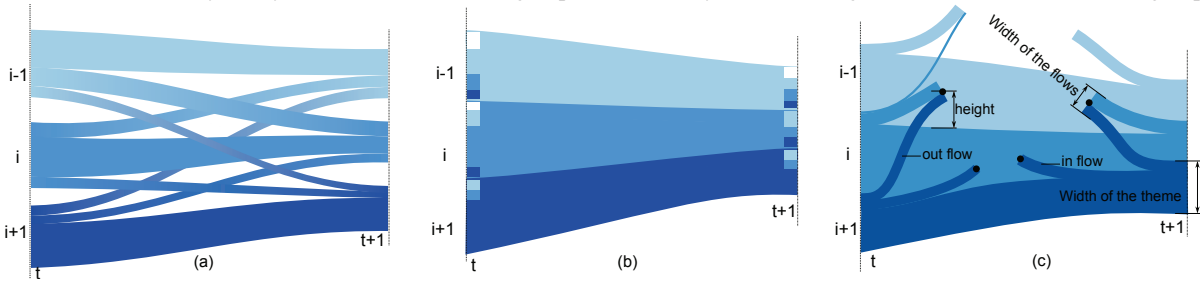


Fig. 4: (a) TextFlow based on a flow-based metaphor, (b) RankExplorer based on color bars, and (c) The flow view based on a new flow metaphor.

- Each layer in the flow view represents a certain degree of user loyalty. The width of the layer at a time point indicates the respective number of the users at that time.
- The branches represent the users that flow in or out of the layers. The width of a branch encodes the number of users. Each cell contains two types of branches: the outflow branches on the left and inflow branches on the right (see Fig. 4(c)).
- The branches of the same type (e.g., inflow) in a cell are bundled to create a neat layout. The bundling order is determined by the order of the layers that the branches flow from (inflow) or to (outflow). The vertical position of a bundled group encodes the average loyalty level of users.
- The branches in the empty region above indicate new users who have just started using the search engine, or leaving users who stop using the current search engine.
- We place a switching histogram above the flow design to provide an overview of user switching behavior at each time point. The color indicates where the new users come from or where the leaving users go to (see Fig. 1). Flow based visualization could also show the user switching behavior across multiple search engines. However, after some experiments, we found that this solution suffered from severe visual clutter and can hardly handle comparative analysis and exploration with multiple search engines.

The layout algorithm is described in details as follows.

### 5.1.3 Layout Algorithm

We follow four design principles, including ***legibility***, ***aesthetics***, ***neatness***, and ***faithfulness***, so as to create a good layout. Legibility and aesthetics principles have been commonly employed in creating engaging, storytelling visualizations [2, 35]. Our work employs the ***legibility principle*** to create a proper layout, such that we can clearly reveal the user flow information (design rationality A). Specifically, we avoid putting the branches in a small region. A symmetric visual design is often utilized to enhance the ***aesthetics principle*** [2, 13] to generate an engaging visualization (design rationality B). Moreover, prior research proves that symmetrical objects can be perceived much more strongly [39]. Thus, we enforce the inflow and outflow branches to be symmetrical at each time point (see Fig. 5(a) and (d)).

The neatness and faithfulness principles ensure that we can effectively and faithfully reveal the information in the layout. The ***neatness principle*** is used to reduce visual clutter by aligning objects in an organized manner, because excess and disorganized items usually degrade visual task performance [37]. Thus, we align the endpoints of outflow branches to the right (Fig. 5(a), (c), and (f)), and align the endpoints of inflow branches to the left (Fig. 5(b), (g), and (h)) in each cell.

The ***faithfulness principle*** ensures that the vertical position of each group of bundled branches should be unchanged, considering that the vertical position represents the average loyalty level of the branches. Thus, the branches can only move along a virtual track (see the dashed line in Fig. 5(e)) within the cell.

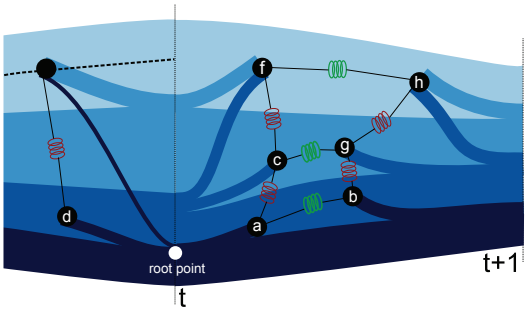


Fig. 5: Illustration for the force-directed model. Black nodes represent the group of the bundled branches. Nodes are connected by springs to achieve our layout goals. The dashed curve in the top left corner shows the track that the node can move along.

Following these principles, a force-directed model is developed to create the layout. We assume that a node placed in the middle of the endpoint of the group of bundled branches (the black nodes in Fig. 5) can represent the group. The model has three basic forces: a spring force, a repulsive force, and a symmetric force.

To meets the legibility principle, a spring force (the green springs in Fig. 5) is used to cause attraction between the inflow and outflow branches in a cell, which enables the branches to have proper lengths to be legible. The spring force between nodes $a$ and $b$ (see Fig. 5) is defined as follows

$$\mathbf{f_s}(a,b) = k_s(||\mathbf{p_a} - \mathbf{p_b}|| - l) \times (\mathbf{p_a} - \mathbf{p_b})/||\mathbf{p_a} - \mathbf{p_b}|| \qquad (1)$$

where $k_s$ is a given weight, $\mathbf{p_a}$ and $\mathbf{p_b}$ represent the positions of $a$ and $b$, respectively; and $l$ is the original length of the spring. We also employ a repulsive force to make sure that the branches in a cell do not overlap when attractive force is used. The repulsive force follows Coulomb's

law, which describes the electrostatic interaction between electrically charged particles and is defined as:

$$\mathbf{f_e}(a,b) = k_e q_a q_b (\mathbf{p_a} - \mathbf{p_b}) / ||\mathbf{p_a} - \mathbf{p_b}||^3 \quad (2)$$

where $k_e$ is the physical constant, and $q_a$ and $q_b$ are the charge of the nodes determined by the widths of the branches.

The spring force is also used to create a neat layout and align the nodes of the outflow branches to the right or the nodes of the inflow branches to the left across the layers (red springs in Fig. 5).

We define a symmetric horizontal force for the inflow and outflow branches sharing the same root in the same layer at a time point (see Fig. 5 (a) and (d)) to satisfy the aesthetics principle.

$$f_{sym}(a) = k_{sym}(x_t + (x_a - x_d)/2 - x_a) \quad (3)$$
$$f_{sym}(d) = k_{sym}(x_t - (x_a - x_d)/2 - x_d) \quad (4)$$

where $k_{sym}$ is a given weight, $x_a$ and $x_d$ represents the positions of nodes $a$ and $d$ along the horizontal axis, respectively, $x_t$ is the horizontal position of the root point of $a$ and $d$ at time point $t$.

When the algorithm starts, in each cell, the branches are placed along the horizontal axis. The algorithm then aggregates all the forces for each node and iteratively moves the particles based on the aggregated force. To satisfy the faithfulness principle, each node can only move along the virtual track within the time frame. This process repeats until the entire particle system achieves a stable state. Notice that all principles are equally important and thus $k_e$ and $k_{sym}$ are assigned appropriately to reflect the property.

#### 5.1.4 User Flow

Tracing the mechanism of flow of a selected group of users across layers over time in the flow view is very important in analyzing user behavior (design rationality $A$). Therefore, we use a design similar to FlowMap [25] to visually track the continuous evolution of a group of selected users in the flow view.

One simple method to show this piece of information is by directly drawing a set of flows in the flow view to show the distribution of the users (see the highlighted orange flows in Fig. 6(a)). The width of a flow encodes the number of the users in a layer or branch. Each flow is initially drawn in the middle position in a branch (i.e., *branch flow*, representing the users switching their category) or in a layer (i.e., *layer flow*, representing the users staying in the same loyalty category between two consecutive time points) (see Fig. 6(a)) to simplify the problem. Although the user distribution is shown, the disconnected flows prevent the analysts tracking the flow patterns readily.

Therefore, we smoothly link the flows together. In a branch (see Fig. 6(a)) in layer $i$, we move down the ***root point*** of the branch flow as small as possible to join the layer flow (below the branch in layer $i-1$) representing the users staying in the same layer. We also extend the ***end point*** of the branch flow to join the layer flow in the current layer $i$. The dashed red line in Fig. 6(a) shows the adjusted flow. This process is repeated for each branch, such that both ends of all the branches join in the layer flows (in neighboring layers $i$ and $i-1$).

To avoid ambiguity, we should make sure that a layer flow is in parallel with the layer, and should not deviate from the middle of the layer too much. Otherwise, the flow may show a wrong increasing or decreasing trend in the layer, which can mislead analysts. We design a method to optimize the positions of flows layer by layer. For layer $i$, we initially select the maximum layer flow and set its position to be the middle of the layer (see the yellow layer flows highlighted in Fig. 6(b) in each layer). Next, we adjust other flows to link them smoothly to the maximum flow in layer $i$ and obtain an initial layout (Fig. 6(b)). Our goal is to find a vertical offset $d$ to adjust the initial layout to minimize the ambiguity. We define the cost as follows.

$$C_i(d) = \sum_{j=1}^{n_l} w_{l_j}(y_{l_j} - y'_{l_j} - d)^2 + \sum_{k=1}^{n_b} w_{b_k}(y_{b_k} - y'_{b_k} - d)^2 \quad (5)$$

where $n_l$ and $n_b$ indicate the number of layer flows and branch flows, respectively; $y$ and $y'$ denote the original (Fig. 6(a)) and initial (Fig. 6(b)) vertical positions, respectively; $w_{l_j}$ and $w_{b_k}$ represent the widths of the branches. We can easily derive the optimal offset $d$ satisfying $C'(d) = 0$. Fig. 6(c) shows the adjusted smooth user flows from Fig. 6(b).

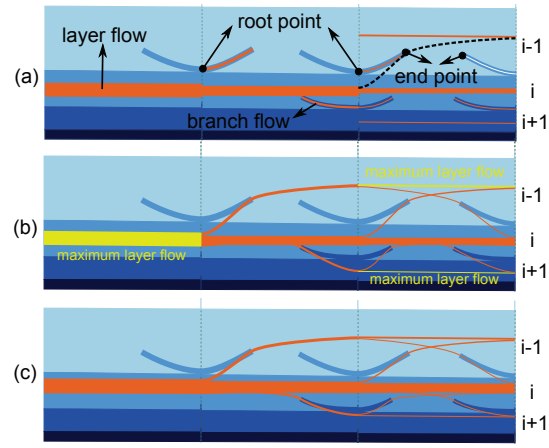


Fig. 6: Illustration of generating the user flow. (a) Straightforward method that shows the distribution of users over time, (b) the initial layout created by fixing the layer flow with the maximum width and adjusting other flows accordingly for the smoothing purpose, and (c) optimal result by minimizing the total cost $C$.

### 5.2 Density Map and Word Cloud

We use two additional visualizations, density map and world cloud, to facilitate the in-depth analysis of user loyalty (design rationality $D$). One important task of loyalty analysis is to investigate the relationship between user loyalty and satisfaction. We use a kernel density estimation (KDE) technique to create a density-based scatterplot called ***density map*** that shows the relationship (see Fig. 9 middle). KDE is particularly effective in plotting large datasets in scatterplots to circumvent the overdraw problem and achieve a truthful assessment of distributional data characteristics.

Our collaborators are also interested in examining the keywords to trigger an engine switch. Thus, we also introduce a visualization (see Fig. 9) to visually summarize the keywords that trigger the switching behavior. Word clouds are chosen because of the excellent storytelling and engaging capability [38] (design rationality $B$). Additionally, we place a color bar under each keyword to intuitively reveal the distribution of different switching types, as requested by our collaborators (See Fig. 9). Each color bar under a word is divided vertically into multiple parts. Each part represents one type of engine switches and contains two color blocks (see the legend above the word cloud in Fig. 9): the upper color encodes the previously used search engine and the lower color encodes the newly switched search engine. Also, the ratio between the size of each part and the total size of the color bar visually encodes the percentage of each type of engine switches.

### 5.3 User Interactions

Design rationality $D$ requests that the system enables analysts to interact with the data. Apart from basic interactions such as pan and zoom, the system supports a set of user interactions.

**Multi-scale exploration** is supported by LoyalTracker to facilitate visual detection and analysis of patterns at different levels of detail (design rationality $C$). LoyalTracker enables analysts to select a time period and a scale (day, week, or month). The flow view will be updated automatically when the analysts choose a new time period or a scale.

**Comparative visualization** can help analysts identify the similarity and difference between multiple search engines from the perspective of dynamic loyalty variation (design rationality $E$). Our system naturally enables interactive comparative visualization by showing multiple linked LoyalTrackers (see Fig. 10).

**Filtering** enables analysts to focus on important information. In a LoyalTracker, not all branches are important. Generally, the larger the

branches, the more important they are. Our system allows an analyst to interactively remove less important branches.

**Brushing** enables analysts to interactively choose their interested data in the visualization directly for further analysis. All visualizations are linked to allow for data exploration from different perspectives. An analyst can easily specify a fairly complex query by brushing multiple branches or layers in the graph. The branches are then combined by taking the union or intersection of the branches, thus allowing the analyst to visually construct queries in a very flexible manner.

## 6 Evaluation and Discussion

We implemented the system using Java. After data preprocessing, interactive performance can be achieved using a PC with Intel(R) Core(TM) i7-2600 CPU and 8 GB memory. To further evaluate LoyalTracker system, we conducted interviews with five domain experts.

### 6.1 Comparison with Munroe's Chart

In the experiment, we compare Munroe's chart [24] with the result of our automatic method. We extracted a part of the original chart (see Fig. 3 left) and then manually labeled the data used for our experiment. Fig. 3 right shows the result. Although the data we used is not exactly the same with the original one (we were unable to obtain the same data and thus only estimated the width of the branches in the chart), we can find that our result can preserve the original information in Munroe's chart. Both figures enable an analyst to easily trace and identify the dynamics of the evolution of different political groups. Compared with the original chart, our result appears clean, neat, and simple.

### 6.2 Case Studies

We extracted the top 100,000 active customers in the US market during the first week of July 2012 from the massive search logs, as suggested by the domain experts. We then acquired the search behavior data on these users from July 2012 through December 2012. The domain experts were allowed freely to classify users into different categories according to the loyalty levels of the users. Based on their background knowledge, after several trials, they settled down four categories: *more than 26, 16-25, 8-15, and less than 7* for the weekly level, or *more than 6, 5, 3-4, and 1-2* for the daily level.

#### 6.2.1 Tracing Customer Loyalty in Individual Engines

The first case study was to demonstrate the usefulness of LoyalTracker on analyzing the dynamics of user loyalty in engine **A**.

Choosing a time scale and a segmentation is the first step for analysts to explore data using LoyalTracker. In this step, they can try different scales in order to see both long-term behavior and short-term behavior. In this case study, we explored the data using two scale levels: daily level (Fig. 7) and weekly level (Fig. 1). The flow views can provide an overview of the dynamics of user loyalty.

From the flow view at daily level (Fig. 7), we can clearly see a pattern: many customers did not use engine **A** on the weekend. From the flow view at weekly level (Fig. 1), we can see no significant change in the sizes of different loyalty categories over time. The inflow and outflow branches have nearly the same size at each time point, indicating that there are almost equal numbers of customers entering or leaving a layer at each time point. Thus, each layer appears flat over time. Additionally, the figure also reveals that most customers tend to change their loyalty level between adjacent layers. Given that the vertical position of the endpoint of a branch inside a layer encodes the average level of customer loyalty, we see that most branches lie in lower parts of the layers, indicating that the changes in loyalty are usually slight. The stable nature of customer loyalty has also been reported in prior research [27].

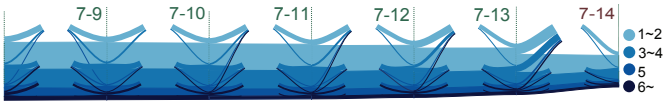


Fig. 7: The flow view shows the loyalty change of the engine **A** at the daily level. Many customers stop using it on the weekend (July 14th).

From the switching histogram (Fig. 1), we can see the statistics of engine switching behavior. More users who left current engine **A** switched to engine **C** than to engine **B**. By tracking the overall height in the flow view, we can observe that the number of total users is decreasing gradually. After examining all the engines, we found that the decreasing number was not caused by engine switching, given that all search engines exhibited the same phenomenon. The domain experts identified two possible reasons: 1) customers started to use other Web browsers; 2) customers cleaned the cookies of the browser. In such circumstance, we cannot track them any more.

When a proper time scale and a suitable segmentation are selected, analysts can further interactively explore the data. User flow is the major interactive visualization component in the flow view, which can help analyze the loyalty dynamics for a specific group of users selected by the brushing interaction.

To better understand the behavior of users with different loyalty levels, we studied two groups of users in the flow view: one group of users stably staying in a cell (layer flow) and the other switching from one layer to its upper layer (branch flow). We visually traced how both groups distribute across layers over time in the engine by drawing user flows highlighted in orange in Fig. 1 top (layer flow) and bottom (branch flow). We can clearly observe an interesting pattern. Compared with the user flow (top), the bottom user flow reveals that once the loyalty of users decreased (i.e., the users flowing through the outflow branch), only a few users would increase their loyalty again. In contrast, the top user flow appears more uniformly distributed.

With the density map and the word cloud view, analysts can obtain more detailed information about the user behavior of the selected group of users. We visually examined the relationship between satisfaction and loyalty using the density map. We selected two groups of users in the same layer at the same time point. One group is in the layer, whereas the other group is in the branch (see Fig. 1). We compared the density maps for the two groups of users (Fig. 8). Surprisingly, the figures show no significant difference between the density maps. It is not an isolated case: across all layers and in all the three engines, we were unable to identify any strong correlation between loyalty and satisfaction based on the density map.

The interaction with the density map enables analysts to further select users according to both satisfaction and loyalty. We selected a group of users with the lowest loyalty (from the top layer in Fig. 1) and then examined the relationships between user loyalty and satisfaction. The density map shows that many users were quite satisfied but had low loyalty (Region ***a*** in Fig. 9). Our domain experts initially speculated that the users could have just conducted navigational searches (i.e., a user searches the name of another search engine in search engine **A** and then switches to other search engines). We brushed different regions in the density map and examined the associated word clouds. By comparing the word cloud of Region ***a*** (see Fig. 9 left) with those of other regions (for example, the word cloud of Region ***b*** in Fig. 9 right), we found that the ratio of navigational search in other regions is higher than that in region ***a***. This pattern can be found consistently from the

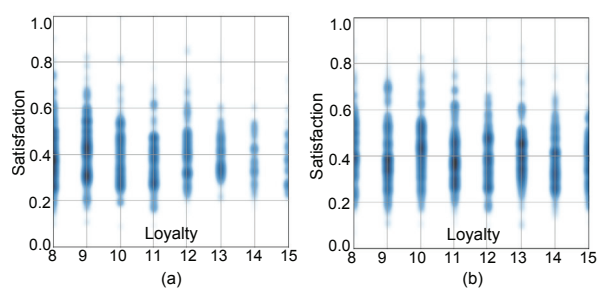


Fig. 8: Density maps (a) and (b) show the relationship between loyalty (horizontal axis) and satisfaction (vertical axis) of the selected group of users in the branch and the layer in Fig. 1, respectively.

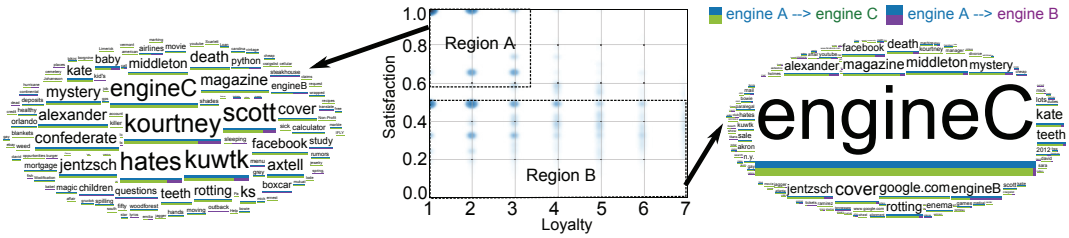


Fig. 9: Density map shows the relations between loyalty and satisfaction for a user group who are less loyal to search engine **A**. The word clouds on the left and right show the keywords triggering engine switching for the users in Region A and Region B of the density map, respectively.

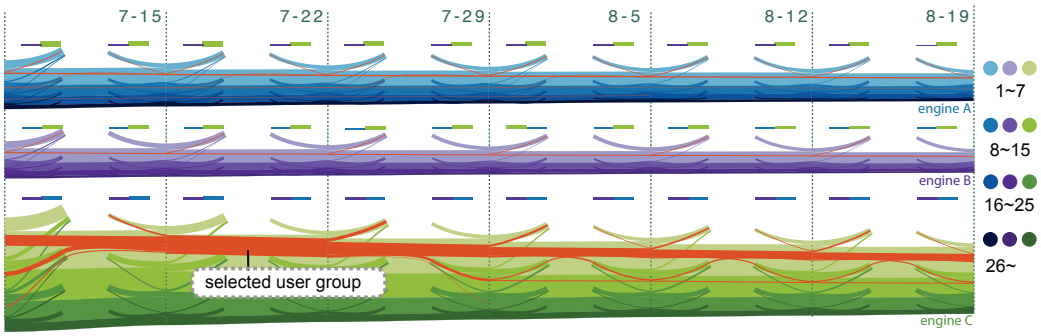


Fig. 10: The flow views for all the three search engines. The width of the customer indicates search engine **C** has the most number of users. The user flow shows the distribution of the selected user group in engine **C**. From the switching histograms, we can see that the total ratio of the customers who left **C** to other search engines is much lower than that in engine **A** and **B**.

top layer (representing the least loyal user) in **A**. However, this pattern cannot be found in any other search engines. The experts suggested some possible reasons. First, our satisfaction measure determines that a navigational search is an unsatisfied query because users do not stay in the search engines for more than 30 seconds. Thus, the users were treated as unsatisfied. Second, it was possible that **A** is set as the default search engine in the widely-used web browser and the users using the browser would perform more navigational search from **A** to others.

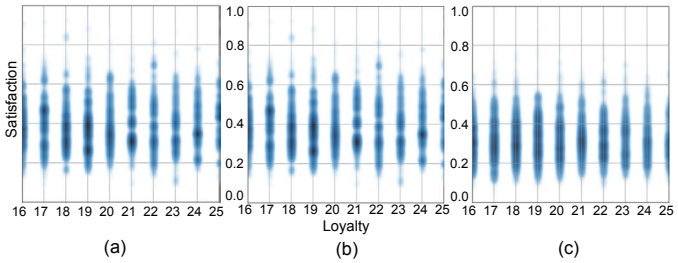


Fig. 11: Density maps (a), (b), and (c) show the relation between loyalty and satisfaction for user groups in the same layers at the same time point on the engine **A**, **B**, and **C**, respectively.

### 6.2.2 Tracing Customer Loyalty in Multiple Engines

The second case study demonstrated the use of our visualization techniques in comparative analysis for multiple search engines (Fig. 10). The overall heights in the three flow views directly indicate the popularity of the three search engines among the active users: more users employ engine **C** than the other two engines. The number of users of engine **A** is slightly larger than engine **B**.

From the switching histograms for the three search engines, we can clearly see that there is a different pattern for engine **C** compared with engine **A** and **B**: the total ratio of the users who left **C** to other search engines is much lower than that in engine **A** and **B**.

By selecting a group of users, we can simultaneously see the user flows in all engines. Fig. 10 shows the user flow of the user group selected from engine **C**. A few selected users also used the other two search engines (**A** and **B**) at the same time, but with very different loyalty distributions: the users only exist in the first layer of engine **B**, whereas they exist in all layers except the bottom one of engine **A**.

We then examined the relationship between loyalty and satisfaction for the three engines. By selecting the users in the same layers at the same time point in three engines, we obtained the corresponding three density maps shown in Fig. 11. We compared the three density maps and found that despite being much more popular and having the largest number of users, search engine **C** does not exhibit a competitive advantage over its competitors in terms of user satisfaction. It is an unexpected pattern. Obviously, the users in search engine **C** are less satisfied compared with **A** and **B**. The domain experts pointed out two possible reasons: 1) users who use search engine **C** did more long tail/ rare queries. It is often difficult to get good results from the search engine [32]; 2) the satisfaction metric should be improved. For example, in many cases users can acquire the information they want on the result page without clicking the links of the results. However, in this case, the corresponding queries are classified as unsatisfied queries.

## 6.3 Interview with Domain Experts

We conducted in-depth interviews with five experts to evaluate the usability of the system, including four program managers (PMs) and a software development engineer (SDE) from the search department of a company. The PMs are particularly interested in analyzing user behavior patterns from search logs. They use *XSystem* daily to analyze the search logs. The SDE gathers feedback about *XSystem* from the users and maintains the system. Each interview lasted for 1.5 - 2 hours. It started with a few questions to identify their background, followed by a tutorial to show the system features. We then asked them to freely explore a few data sets within the system. We finally asked them some post-study questions to collect their feedbacks and suggestions. We denote the participants as PM1, PM2, PM3, PM4, and SDE.

***Overall system usability.*** The system was received very well by all participants. PM4, the most senior PM, pro-actively contacted us and requested to try our system when she learned about it from other PMs. She commented "*The system has a great potential and is powerful for finding and analyzing user switching patterns.*". Both PM2 and PM3 commented that the system is capable of helping them evaluate the effectiveness of a newly added feature of the search engine. All PMs were keen on using the comparative visualization feature provided by the system. SDE emphasized that the system would be valuable for the users of *XSystem* that is heavily used by a few hundreds of project managers, applied scientists, and other analysts in the company.

***Visual design and interactions.*** All participants were impressed by the visual design and the supported interactions. They especially like the narrative nature by LoyalTracker that can help reveal dynamic user loyalty variation visually and intuitively. PM2 felt very excited about the visual design and commented "*I can clearly see the macro overall trend of user loyalty variation as well as the micro trend of user flowing patterns in one view*". They accepted the concept of multi-scale exploration very well and agreed that the switching histogram is intuitive to understand and useful for showing switching behavior.

The participants appreciated the interactions supported by the system. They acknowledged the usefulness of the filtering and brushing interactions. PM2 said "*the user interactions would greatly facilitate my analysis tasks*". PM2 and PM4 commented that user flow enables them to easily connect the users who share a similar loyalty pattern and trace their loyalty variation across multiple search engines.

The participants also acknowledged the usefulness of the density map and the word cloud. PM1 and PM2 commented that the word cloud can provide them a quick overview of the distribution of the words that trigger engine switching behavior. PM4 liked the density map and commented that "*it allows me to quickly see the relationship between satisfaction and loyalty for a group of users*". All participants agreed on the usefulness of the linked visualizations.

***Suggestions.*** The participants provide valuable feedback about the system. All PMs except for PM3 suggested that we should support more data filtering operations, such as the filtering based on different entry points (such as tool bars or homepages). All PMs also highlighted the potential of the system to support other in-depth analysis. PM2 particularly commented that "*The visualization is good at enabling qualitative analysis. It would be desired that quantitative analysis can also be supported*". The participants also had some concerns. PM1 and PM4 had concerns about the intuitiveness of the system. While it was easy for them to understand each individual view, they felt difficult to link them together. Nevertheless, they both agreed that after a short time training they could get used to the linked system.

### 6.4 Discussion

The search engine logs exhibit rich and valuable user information, which allow us to acquire a better understanding of user loyalty analysis and derive a set of considerate design principles. Although we mainly demonstrate our visualization techniques using search engine logs, the techniques could be easily adapted to other problems, such as user engagement on e-learning courses, which share similar data characteristics and task requirements.

We employ a force-directed layout algorithm based on different effectiveness and aesthetics criteria to generate a layout. The layout algorithm works well when most branches do not cross more than two layers. It may fail to create an effective layout in some extreme cases in which there are frequent and dramatic exchanges of quantities across the distant layers. In this case, there would be a significant amount of clutter caused by crossings between the branches and the layers. The visual design based on color bars [30] would be helpful in these scenarios. Nevertheless, we believe that our design works for many scenarios of customer loyalty analysis as the loyalty degree of a customer will not dramatically change in most cases [27].

From the feedback from the expert review, the flow view layout is easy to be understood. One typical benefit of it is to allow an analyst to interactively specify a fairly complex query to find a group of users based on the user flow. However, the brushing interactions across multiple linked views may cause some learning costs, as we discussed in Section 6.3.

In the case study, we visually compared three search engines using our system. It can naturally support more search engines by adding more flow views. But due to the limited screen space and capability of users for comparison tasks [47], the comparison for more than four engines are not recommended. In LoyalTracker, we offer one density map view and one word cloud view. When conducting comparative analysis, it poses a challenge for analysts to visually compare multiple word clouds or multiple density maps. However, the smooth and responsive user interaction can largely help circumvent this problem. Analyst can quickly brush and see the results immediately. The case studies and expert reviews also confirm the usability and effectiveness of current interface and interaction design. In the flow view, we use different colors to encode the categories of loyalty. According to the previous study [18] and the suggestion from our experts, we classify the users into four groups. The flow view allows more than four categories. However, as people can efficiently distinguish only a dozen colors [39], the number of the loyalty categories should be less than twelve.

The user log data we used also has some limitations:

- **Uncertainty**. Users are recognized by the ID stored in the cookie of a web browser in the data. Thus, when users change to another web browser or clean the cookies, we cannot track them any more. Furthermore, the user log data assumes one web browser in one computer used by only one person, which is not always the case. Although all the logs are stored anonymously, it is feasible to infer if two users are the same or not using some data fields such as IP address or the clicking behaviors, which could introduce uncertainty to the data. Some visual hints can be displayed to keep user aware of the uncertainty.
- **Scalability**. The data processing part cannot be done real-time. Nevertheless, after that LoyalTracker can achieve interactive performance. From the case study, we found that explaining some patterns needs more original raw data from the data center. Due to the large scale of the data and the limited computing resource we have, in the current stage we cannot retrieve the raw data in real time. However, this problem can be solved in the future by applying in more computing resource.

## 7 Conclusion and Future Work

In this paper, we systematically study the effective visualization of customer loyalty and switching behavior from massive data sets in business intelligence. We derive a set of design principles to address the most important questions raised by the domain experts of loyalty analysis. Guided by the principles, we propose a visual analytics system. The system consists of three views: a flow view, a density map view, and a word cloud view. We design a new visualization technique in the flow view, based on a flow metaphor to interactively reveal the evolving patterns of customer loyalty and defection. The other two views: a density map view and a word cloud view enable in-depth analysis.

Case studies and the interview with domain experts demonstrated the usefulness of the flow view for showing the overall loyalty trend along the time. The flow view plays a primary role in discovering the patterns we just discussed. It enables analysts to quickly and intuitively construct a fairly complex queries, such that they can perform detailed analysis and confirm the findings in the density map and word cloud.

In the future, we plan to release a web based version of LoyalTracker to the search technology department of the company with a larger number of potential users. We are also planing to improve the interaction design in order to reduce the learning curve and make the system easier to use. For example, to reduce the complexity of the brushing interaction, more visual hints will be used to mark analysts interactions (i.e., union operations and intersection operations) so that the brushing results can be easily understood. Also, when analysts explore multiple flow views, synced zoom & pan operations will be offered. Moreover, we plan to apply our techniques to other similar problems. For example, we can use them to track and analyze the dynamics of user engagement in e-learning courses from web log data. Thus, our techniques can be used by a large group of audience, which enable us to get richer feedback and suggestions. We will also further evaluate our techniques using a formal user study. Visualization of the co-evolution pattern of loyalty and satisfaction is another potential future research direction.

## Acknowledgments

The authors wish to thank Aviz group in Inria, France, for their kind help on revising the paper and the anonymous reviewers for their valuable comments. This research was supported in part by HK RGC GRF 16208514, the National Basic Research Program of China (973 Program) under Grant No. 2014CB340304, the National Natural Science Foundation of China 61103055, and FDYT LYM11113.

## REFERENCES


[1] E. W. Anderson and V. Mittal. Strengthening the satisfaction-profit chain. *Journal of Service Research*, 3(2):107–120, 2000.

[2] L. Byron and M. Wattenberg. Stacked graphs – geometry & aesthetics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1245–52, 2008.

[3] S. K. Card, P. Pirolli, M. Van Der Wege, J. B. Morrison, R. W. Reeder, P. K. Schraedley, and J. Boshart. Information scent as a driver of web behavior graphs: results of a protocol analysis method for web usability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 498–505, 2001.

[4] E. H. Chi. Improving web usability through visualization. *IEEE Internet Computing*, 6(2):64–71, 2002.

[5] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. J. Gao, X. Tong, and H. Qu. Textflow: towards better understanding of evolving topics in text. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2412–2421, 2011.

[6] M. Dörk, D. M. Gruen, C. Williamson, and M. S. T. Carpendale. A visual backchannel for large-scale events. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1129–1138, 2010.

[7] G. Dupret, V. Murdock, and B. Piwowarski. Web search engine evaluation using clickthrough data and a user model. In *WWW workshop Query Log Analysis: Social and Technological Challenges*, number April.

[8] M. Gray, A. N. Badre, and M. J. Guzdial. Visualizing usability log data. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 93–98, 1996.

[9] Q. Guo, R. W. White, Y. Zhang, B. Anderson, and S. T. Dumais. Why searchers switch: understanding and predicting engine switching rationales. In *Proceedings of the ACM SIGIR conference on Research and development in information retrieval*, pages 335–344, 2011.

[10] M. Guzdial, P. Santos, A. Badre, S. Hudson, and M. Gray. Analyzing and visualizing log files: A computational science of usability. Technical report, Georgia Institute of Technology, 1994.

[11] M. Hagen, M. Potthast, B. Stein, and C. Bräutigam. Query segmentation revisited. In *Proceedings of the 20th international conference on World wide web - WWW '11*, page 97, New York, USA, Mar. 2011.

[12] A. Hassan, X. Shi, N. Craswell, and B. Ramsey. Beyond Clicks: Query Reformulation as a Predictor of Search Satisfaction. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 2019–2028, New York,USA, 2013.

[13] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.

[14] V. Hu, M. Stone, J. Pedersen, and R. W. White. Effects of search success on search engine re-use. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, page 1841, New York, New York, USA, Oct. 2011. ACM Press.

[15] J. Huang and E. N. Efthimiadis. Analyzing and evaluating query reformulation strategies in web search logs. In *Proceeding of the 18th ACM conference on Information and knowledge management*, page 77, New York, New York, USA, Nov. 2009. ACM Press.

[16] Y.-F. Juan and C.-C. Chang. An analysis of search engine switching behavior using click streams. In *Proceedings of the First International Workshop on Internet and Network Economics*, pages 806–815, 2005.

[17] S. M. Keaveney and M. Parthasarathy. Customer switching behavior in online services: An exploratory study of the role of selected attitudinal, behavioral, and demographic factors. *Journal of the Academy of Marketing Science*, 29(4):374–390, 2001.

[18] P. Kotler and K. Keller. *Marketing Management*. Prentice Hall, 14th edition, 2011.

[19] H. Lam, D. Russell, D. Tang, and T. Munzner. Session viewer: Visual exploratory analysis of web session logs. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 147–154, 2007.

[20] J. Lee, M. Podlaseck, E. Schonberg, and R. Hoch. Visualization and analysis of clickstream data of online stores for understanding web merchandising. *Data Mining and Knowledge Discovery*, 5(1/2):59–84, 2001.

[21] S. Liu, W. Cui, Y. Wu, and M. Liu. A survey on information visualization: recent advances and challenges. *The Visual Computer*, pages 1–21, 2014.

[22] S. Liu, M. X. Zhou, S. Pan, W. Qian, W. Cai, and X. Lian. Interactive, topic-based visual text summarization and analysis. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 543–552, 2009.

[23] T. Mukhopadhyay, U. Rajan, and R. Telang. Competition between internet search engines. In *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2004.

[24] R. Munroe. A history of the united states congress – partisan and ideological makeup. http://xkcd.com/1127/.

[25] D. Phan, L. Xiao, R. B. Yeh, P. Hanrahan, and T. Winograd. Flow map layout. In *Proceedings of IEEE Symposium on Information Visualization*, pages 219 – 224, 2005.

[26] J. Pitkow and K. A. Bharat. Webviz: A tool for world-wide web access log analysis. In *Proceedings of the International World Wide Web Conference*, pages 271–277, 1994.

[27] F. F. Reichheld and P. Schefter. E-Loyalty: your secret weapon on the web. *Harvard Business Review*, 78(4):105–113, 2000.

[28] F. F. Reichheld and J. W. Earl Sasser. Zero defections: quality comes to services. *Harvard Business Review*, 68(5):105–113, 1990.

[29] Z. Shen, J. Wei, N. Sundaresan, and K.-L. Ma. Visual analysis of massive web session data. In *Proceedings of the IEEE Symposium on Large Data Analysis and Visualization*, pages 65–72, 2012.

[30] C. Shi, W. Cui, S. Liu, P. Xu, W. Chen, and H. Qu. RankExplorer: Visualization of ranking changes in large time series data. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2669–2678, 2012.

[31] L. Shi, F. Wei, S. Liu, L. Tan, X. Lian, and M. X. Zhou. Understanding text corpora with multiple facets. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 99–106, 2010.

[32] Y. Song and L.-w. He. Optimal rare query suggestion with implicit user feedback. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 901–910, New York, NY, USA, 2010. ACM.

[33] Y. Song, X. Shi, and X. Fu. Evaluating and predicting user engagement change with degraded search relevance. In *Proceedings of the International World Wide Web Conference (To appear)*, 2013.

[34] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu. A survey of visual analytics techniques and applications: State-of-the-art research and future challenges. *Computer Science and Technology*, 28(5):852–867, 2013.

[35] Y. Tanahashi and K.-L. Ma. Design considerations for optimizing storyline visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2679–2688, 2012.

[36] H. Tsuji and R. Saga. Visualized preference transition network based on recency and frequency. *IEMS*, 10(4):238–246, 2011.

[37] R. van den Berg, F. W. Cornelissen, and J. B. T. M. Roerdink. A crowding model of visual clutter. *Journal of Vision*, 9(4):Article 24, 2009.

[38] F. B. Viégas, M. Wattenberg, and J. Feinberg. Participatory visualization with wordle. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1137–1144, 2009.

[39] C. Ware. *Information Visualization, Third Edition: Perception for Design (Interactive Technologies)*. Morgan Kaufmann, 3rd edition, 2012.

[40] S. J. Waterson, J. I. Hong, T. Sohn, J. A. Landay, J. Heer, and T. Matthews. What did they do? understanding clickstreams with the WebQuilt visualization system. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, pages 94–102, 2002.

[41] J. Wei, Z. Shen, N. Sundaresan, and K.-L. Ma. Visual cluster exploration of web clickstream data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 3–12, 2012.

[42] R. W. White and S. T. Dumais. Characterzing and predicting search engine switching behavior. In *Proceedings of the ACM conference on Information and knowledge management*, pages 87–96, 2009.

[43] R. W. White, A. Kapoor, and S. T. Dumais. Modeling long-term search engine usage. In *Proceedings of the international conference on User Modeling, Adaptation, and Personalization*, pages 28–39, 2010.

[44] R. W. White, M. Richardson, M. Bilenko, and A. P. Heath. Enhancing web search by promoting multiple search engine use. In *Proceedings of the ACM SIGIR conference on Research and development in information retrieval*, pages 43–50, 2008.

[45] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2659–2668, 2012.

[46] Z. Yang and R. T. Peterson. Customer perceived value, satisfaction, and loyalty: The role of switching costs. *Psychology & Marketing*, 21(10):799–822, 2004.

[47] S. Yantis. Multielement visual tracking: attention and perceptual organization. *Cognitive psychology*, 24(3):295–340, July 1992.

[48] J. Zhang, C. Chen, M. S. Vogeley, D. Pan, A. Thakar, and J. Raddick. SDSS Log Viewer: visual exploratory analysis of large-volume sql log data. In *Proceedings of the Visualization and Data Analysis*, 2012.