Information Processing and Management 51 (2015) 114–129

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Building a better mousetrap: Compressing mouse cursor activity for web analytics

Luis A. Leiva [a,*], Jeff Huang [b,1]

[a] PRHLT Research Center, Universitat Politècnica de València, 46022 Valencia, Spain
[b] Computer Science Department, Brown University, Providence, RI 02912, United States

ARTICLE INFO

Article history:
Received 26 December 2013
Received in revised form 16 October 2014
Accepted 22 October 2014
Available online 15 December 2014

Keywords:
Mouse cursor tracking
Web analytics
Temporal-spatial data compression
Data simplification

ABSTRACT

Websites can learn what their users do on their pages to provide better content and services to those users. A website can easily find out *where* a user has been, but in order to find out *what* content is consumed and *how* it was consumed at a sub-page level, prior work has proposed client-side tracking to record cursor activity, which is useful for computing the relevance for search results or determining user attention on a page. While recording cursor interactions can be done without disturbing the user, the overhead of recording the cursor trail and transmitting this data over the network can be substantial. In our work, we investigate methods to compress cursor data, taking advantage of the fact that not every cursor coordinate has equal value to the website developer. We evaluate 5 lossless and 5 lossy compression algorithms over two datasets, reporting results about client-side performance, space savings, and how well a lossy algorithm can replicate the original cursor trail. The results show that different compression techniques may be suitable for different goals: LZW offers reasonable lossless compression, but lossy algorithms such as piecewise linear interpolation and distance-thresholding offer better client-side performance and bandwidth reduction.

1. Introduction

Websites that know what their users do on their pages can provide better content and services to those users. For example, search engines can re-rank search results using what people click as implicit feedback (either personalizing results for individuals from their click history, or using aggregated data from past searches to improve the overall ranking), e-commerce sites can learn what parts of the page deter potential customers, and social networking sites can use aggregated usage metrics to improve the usability of their application. A website can easily find out *where* a user has been on their pages through server access logs, but this yields an incomplete picture of what their users were actually doing. To find out *what* content is consumed and *how* it was consumed at the page level, the website can use client-side tracking to record richer interactions such as cursor movements and hovering, scrolling activity, and text highlighting. These interactions can be interpreted into higher-level behaviors like reading and marking interesting text with the cursor, quickly skimming the entire page, or moving the cursor out of the way to the side. To this end, page-level interactions provide researchers and practitioners with a way to gain additional insight of users' Web browsing behavior.

* Corresponding author. Tel.: +34 963878172.
E-mail addresses: llt@acm.org (L.A. Leiva), ipm@jeffhuang.com (J. Huang).
[1] Work partially conducted at the University of Washington.

http://dx.doi.org/10.1016/j.ipm.2014.10.005
0306-4573/

Previous literature has shown value in using mouse cursor interaction data for applications such as: determining the relevance of search results (Guo & Agichtein, 2012; Huang, White, Buscher, & Wang, 2012b; Speicher, Both, & Gaedke, 2013), to track what users are reading on a page (Diaz, White, Buscher, & Liebling, 2013; Guo & Agichtein, 2010b; Hauger, Paramythis, & Weibelzahl, 2011; Hauger & Van Velsen, 2009), user modeling (Buscher, White, Dumais, & Huang, 2012; Leiva & Vidal, 2010), or potentially as a proxy for gaze tracking (Huang, White, & Buscher, 2012a; Huang, White, & Dumais, 2011; Rodden, Fu, Aula, & Spiro, 2008). Many commercial Web analytics services allow websites to track their users' mouse cursor interactions: ClickTale, LuckyOrange, MouseFlow, Mpathy, and Clixpy. Once set up on a website, the analytics service allows Web developers the ability to replay cursor movements from a user session, or generate heatmaps with aggregated cursor positions.

While recording cursor interactions can be done without interrupting the user, the overhead of recording the cursor movements and transmitting this data to the server can be substantial. For instance, swiping the mouse from left to right can generate a hundred {x, y} cursor coordinate pairs, which over a minute of interaction can lead to nearly 1 MB of data being sent to the analytics server. Compressing cursor activity is challenging because current transport protocols have no mechanism for compressing this type of high-frequency client-side generated data. In our work, we investigate methods to compress the trail a cursor makes on a page, taking advantage of the fact that not every cursor coordinate has equal value to the web developer. This problem is particularly important for situations where bandwidth is limited, such as on a mobile network or in developing countries. And while there are a limited number of services offering cursor tracking features, having efficient cursor tracking methods will benefit all Web users. One Web analytics company may be used by a thousand websites which in turn may serve a million users.

The contribution we make in this paper is a rigorous evaluation of 10 compression algorithms for 3-dimensional data (x-coordinate, y-coordinate, and time), evaluated with 4 metrics that represent different needs for the compressed data. The evaluation is conducted across both a dataset collected from a lab study and a dataset from a live web page, both datasets involving real users. We show that different compression techniques may be useful in different situations; the situations can reflect a desire for consuming less bandwidth, better client-side performance, more accurate replication of the original data, or a combination of all three objectives. With the reduction in data size, tracking mouse cursor interactions can finally become a practical, scalable technology. As a secondary contribution, we share the dataset we collected and the implementations of the compression algorithms we used in our experiments.[2] This will allow commercial analytics services to build on top of our work, and for other researchers to replicate our results.

2. Related work

Mouse cursor activity has been applied to different applications in prior work. An early study by Chen, Anderson, and Sohn (2001) found a relationship between cursor positions and where people were looking, suggesting a potential for using cursor tracking to substitute eye-tracking. Hauger et al. conducted a study over instructional pages about the game of "Go" (Hauger et al., 2011). They found that gaze and cursor positions were better correlated when the cursor was in motion and in sessions comprising a higher proportion of motion. Hauger et al. were able to predict which paragraph a user was reading, and to what extent, with 79% accuracy. Guo and Agichtein show some utility of the cursor position as a proxy of where a user is looking (Guo & Agichtein, 2010b) but Huang et al. (2012a) caution against assuming that the cursor approximates eye-gaze, as this is often not the case depending on time and the user's current action.

In information retrieval, cursor tracking can help determine the relevance of search results (Guo & Agichtein, 2012; Huang et al., 2011; Speicher et al., 2013). When a user is focusing on particular results with their cursor, even when not clicking, it may be a signal to the search engine that the search result is relevant to the user. Huang et al. take this further and incorporate cursor hovering and scrolling into a user model that is shown to help label search results for relevance (Huang et al., 2012b). Other researchers have developed user models as well for clustering web documents (Leiva & Vidal, 2010), inferring the reason a user abandons a search (Diriye, White, Buscher, & Dumais, 2012), predicting user attention in novel page layouts (Diaz et al., 2013), restyle the design of a website (Leiva, 2011), determining whether a user's shopping goal (Guo & Agichtein, 2010a), and identifying different types of searchers (Buscher et al., 2012). Additionally, since cursor interactions span a variety of behaviors, a website can learn more about what content attracts their users when the cursor is used to read, highlight, or mark text (Hauger & Van Velsen, 2009; Liu & Chung, 2007; Rodden et al., 2008). Broader interactions may also be interesting to website developers, such as which regions a user hovers over, when the user moves the cursor out of the way, how they fill out forms, or move towards scrolling and clicking (Cooke, 2006; Rodden & Fu, 2007).

2.1. Cursor tracking systems in academia

To our knowledge, Mueller and Lockerd developed the first approach to capture cursor data in the research literature (Mueller & Lockerd, 2001), examining the behavior of 17 participants. However, as an extended abstract, they provide little description of their system architecture, a section comprising two sentences, "Our system posts mouse movement data

[2] See http://hci.cs.brown.edu/mousetrap/.