applied sciences

MDPI

Article

Pattern-Based and Visual Analytics for Visitor Analysis on Websites

Bárbara Cervantes [1], Fernando Gómez [1], Raúl Monroy [1], Octavio Loyola-González [2,*], Miguel Angel Medina-Pérez [1] and José Ramírez-Márquez [3]

[1] Tecnologico de Monterrey, School of Engineering and Science, Carretera al Lago de Guadalupe Km. 3.5, Atizapán, Estado de México 52926, Mexico
[2] Tecnologico de Monterrey, School of Engineering and Science, Vía Atlixcáyotl No. 2301, Reserva Territorial Atlixcáyotl, Puebla 72453, Mexico
[3] Enterprise Science and Engineering Division, Stevens Institute of Technology, School of Systems & Enterprises, Hoboken, NJ 07030, USA
* Correspondence: octavioloyola@tec.mx

Received: 3 July 2019; Accepted: 27 August 2019; Published: 12 September 2019

Featured Application: We present a tool to analyze web log files, complemented by applying pattern mining techniques to characterize segments of users.

Abstract: In this paper, We present how we combined visualization and machine learning techniques to provide an analytic tool for web log data.We designed a visualization where advertisers can observe the visits to their different pages on a site, common web analytic measures and individual user navigation on the site. In this visualization, the users can get insights of the data by looking at key elements of the graph. Additionally, we applied pattern mining techniques to observe common trends in user segments of interest.

Keywords: pattern mining; data visualization; log analysis

1. Introduction

Analyzing and describing visitor behavior of an e-commerce site is of interest to web marketing teams, especially when assessing ad campaigns. Marketing teams are interested in quantifying their human visitors and characterizing them, for example, to discover the common elements of visitors who made a conversion (e-commerce purpose). Also, knowledge about visitor behavior on a website could benefit IT personnel, for example, as an instrument to identify bot visitors and combat click-fraud.

One resource that can be used to analyze visitor behavior is web interaction data, such as mouse and keyboard usage; however, this type of data is not inherently available and requires the collection of information on the visitor side (which often is denied for privacy issues). Conversely, a web log file is a trace available in any server that hosts a website. The requests made by the visitors to the site are recorded in this log file, providing website owners with information about the resources requested, including details about the visitor and the resource itself. Thus, the information from web log files is a valuable asset in the analysis of visitor behavior.

In this work, we present a new visualization that allows web marketing teams to understand how visitors navigate their site, which is key to analyze the success of a campaign or to redesign a website. Our visualization shows the user a general view of the visitors, pages, and their interactions, apart from some common web analytic measures. Moreover, we include a detailed view of a visit navigation path, which allows observing individual behavior. Yet, we do not intend to replace other visualization tools, rather we aim to complement them with interesting features.

Furthermore, in order to help the visual model end-user understand what captures a type of visitor, we apply pattern mining techniques, in particular, contrast patterns. In this case, we are interested in seeing what separates two segments of visitors, so we extract contrast patterns from two classes. This yields patterns which outline what differentiates one segment from the other., enriching our visual model. Features could be used to define classes and form different groups of users that are of interest to characterize. For example, we can apply pattern mining to characterize the traffic that comes from a country of interest against that of the others, or what characterizes visits that yield a conversion. As use case scenarios, we present the characterization of human versus bot visitors, and an example of country segmentation.

A summary of our contributions is as follows:

- The design of an interactive visualization that allows users to have a comprehensive snapshot of visitors on a website, but also enables a fine-grained analysis by means of navigation graphs.
- The application of pattern mining techniques to extract patterns that characterize traffic segments of interest. The obtained patterns can aid in the selection of groups of users whose behavior would be interesting to observe. For example, we have been able to discover patterns that capture groups of interest, including (some types of) human and bot traffic. This last bit is of paramount importance, both because marketing can now give attention to clean and crisp segments of traffic, and because IT may block unwanted traffic, using for example firewall rules.

The paper is organized as follows. First, in Section 2, we present a brief analysis of web analytic tools and summarize research efforts in web visualization. Section 3 describes the data and the pre-processing steps required for the visualization and pattern mining components of our approach. In Section 4, we describe our visualization design. Section 5 describes how this visualization can be enriched by the use of machine learning, specifically of pattern mining techniques. Finally, in Section 6, we discuss the applications of our work and possible extensions.

2. Related Work

There are many tools available for measuring digital content. Nevertheless, they all display web analytics in a similar manner; goal reports, conversions and site performance usually are still displayed as tables, big score counters or line plots. Next, we mention five popular web analytic tools and provide a brief discussion about them. Then, we mention the research proposals related to the visualization of web behavior.

Google Analytics (GA) is Google's main product for getting reports and analyzing the traffic on a website. It can be configured to import and track ad campaigns from Ad Words [1] and Double Click [2] (Google's web advertising products). It allows segmenting the traffic from many sources and by applying several filters. Also, it has the advantage of being widely known by marketing experts and people from other domain areas.

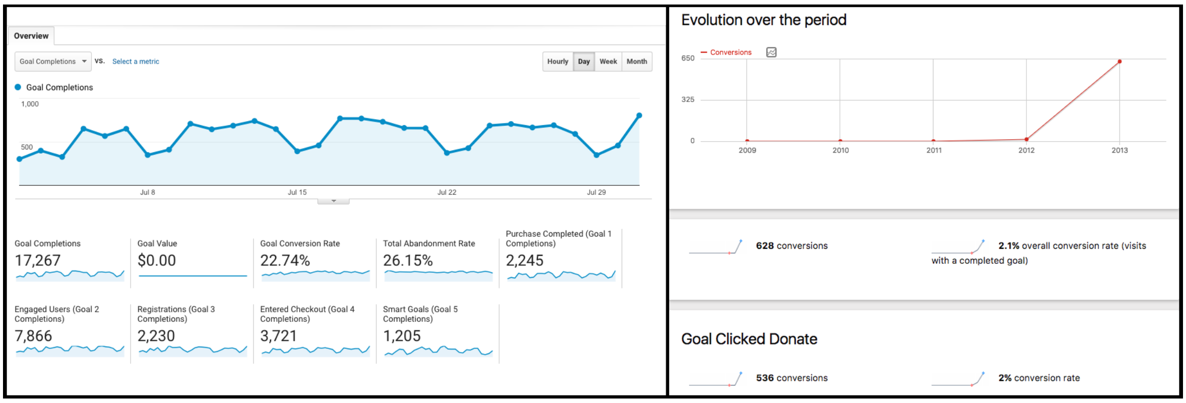
ComScore [3] is an American company which provides services to evaluate media across different platforms, it has a big presence not only on the Internet but also in the TV industry, newspapers, health care, and others. Unfortunately, we could not get a further analysis of their tools because they are paid services. Despite this fact, comScore has been very open with their current research and has been publishing some reports in a periodic way [4,5].

KissMetrics [6] provides analytic reports and email tools to increase user engagement. They provide a more tailored experience, focusing on consulting and teaching their customers on how to configure the tool and interpret the results. It also allows segmenting the traffic using filters. KissMetrics is also a paid service.

Matomo [7], formerly named Piwik, it is one of the most popular and robust tools. It can be self-hosted or as Software as a Service (SaaS) in the cloud. Matomo is a company that focuses on giving their users complete control of everything, meaning that you get full reports (no data sampling, in contrast with GA). It is developed in PHP and also provides an HTTP API for consulting reports such as the visitor's information, goals, and pages performance, user segments, live visits information, among others.

Although the Open Web Analytics (OWA) [8] project has not published any new version since 2014, it is still popular in legacy websites. It was integrated into former versions of Content Management Systems (CMS) like Wordpress or Media Wiki. It can be tested only by installing it on a personal server. One of the features included is a heatmap that shows the hottest (most clicked) sections of a website page, which can be used to optimize the placement of page information.

As mentioned before, many of the web analytics reports provided by the solutions described have not changed significantly in the past few years. Such reports do not provide, for example, how visits interact with the website pages individually; most often, results are aggregated and spread in multiple reports. Figure 1 shows a common report to display goal performance and conversion counters. Although these kinds of reports provide a quick way to compare time series, they could be improved, for example, by integrating information and adding interactive controls.



Goal Completions 17,267

Goal Value $0.00

Goal Conversion Rate 22.74%

Total Abandonment Rate 26.15%

Purchase Completed (Goal 1 Completions) 2,245

Engaged Users (Goal 2 Completions) 7,866

Registrations (Goal 3 Completions) 2,230

Entered Checkout (Goal 4 Completions) 3,721

Smart Goals (Goal 5 Completions) 1,205

Evolution over the period

628 conversions

2.1% overall conversion rate (visits with a completed goal)

Goal Clicked Donate

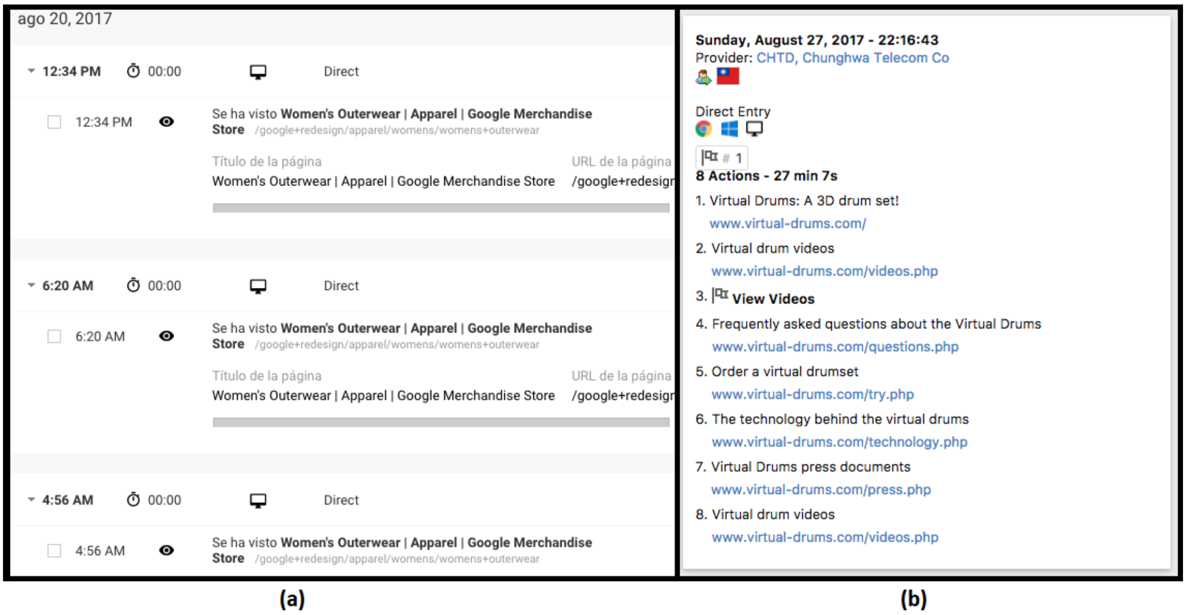
536 conversions

2% conversion rate

(a) **(b)**

Figure 1. Goal reports by Google Analytics (**a**) and Matomo (**b**), typically used to display the performance of user pre-defined goals such as number of sign-ups and purchases through time. (**a**) Screenshot of a GA Day view report obtained from the Google Online Store. (**b**) Screenshot of a Year view report obtained from the Matomo Platform.

Regarding visitor's navigation, the common report is a table of sequential actions or as aggregated traffic flow (funnels). However, when marketers sometimes need to explore individual traces of navigation from their users. Information from individual traces, when available, is represented through simple text reports, as shown in Figure 2. Few platforms have implemented features for automatic customer segmentation. Typically, an expert manually creates filters based on arbitrary parameters such as the visitor's country, user type, visitor's language, among others.; although this is not a problem per se, it is not trivial to select a relevant segment. We believe that machine learning can improve this process by suggesting such filtering parameters through the use of pattern mining [9–11]. Such patterns represent true segments found in the data itself.


ago 20, 2017

▾ **12:34 PM** ⏱ 00:00 Direct

□ 12:34 PM Se ha visto **Women's Outerwear | Apparel | Google Merchandise Store** /google+redesign/apparel/womens/womens+outerwear

Título de la página: Women's Outerwear | Apparel | Google Merchandise Store
URL de la página: /google+redesign

▾ **6:20 AM** ⏱ 00:00 Direct

□ 6:20 AM Se ha visto **Women's Outerwear | Apparel | Google Merchandise Store** /google+redesign/apparel/womens/womens+outerwear

Título de la página: Women's Outerwear | Apparel | Google Merchandise Store
URL de la página: /google+redesign

▾ **4:56 AM** ⏱ 00:00 Direct

□ 4:56 AM Se ha visto **Women's Outerwear | Apparel | Google Merchandise Store** /google+redesign/apparel/womens/womens+outerwear

Sunday, August 27, 2017 - 22:16:43
Provider: CHTD, Chunghwa Telecom Co

Direct Entry

1
8 Actions - 27 min 7s

1. Virtual Drums: A 3D drum set!
   www.virtual-drums.com/
2. Virtual drum videos
   www.virtual-drums.com/videos.php
3. **View Videos**
4. Frequently asked questions about the Virtual Drums
   www.virtual-drums.com/questions.php
5. Order a virtual drumset
   www.virtual-drums.com/try.php
6. The technology behind the virtual drums
   www.virtual-drums.com/technology.php
7. Virtual Drums press documents
   www.virtual-drums.com/press.php
8. Virtual drum videos
   www.virtual-drums.com/videos.php

Figure 2. Browsing history reports by Google Analytics (**a**) and Matomo (**b**). Both are tables of the sequence in which a single visitor was browsing the site. (**a**) Screenshot of a GA user's browsing history. (**b**) Screenshot of a Matomo user's browsing history.

One disadvantage of the enterprise solutions (GA, comScore and KissMetrics) is that they can only be used as a hosted service which can be inconvenient for companies that need an on-premise solution, or if they need to obey certain law regulations about storing customer's data. Another big concern about these solutions is that, usually, the user does not own the data, instead, the only way to access the information is through a third-party. Unlike open source solutions, like Matomo, where the user is the owner of 100% of the data.

Apart from the previously mentioned tools, there have been research efforts towards designing new visualizations. Maps and network graphs are common visualizations, given that traffic comes from any part of the world. For example, work by Akamai [12], provides an interactive map of web attacks in real time. Kaspersky [13], offers a similar tool but using a 3D perspective and more features integrated. Logstalgia [14], is another interesting tool to visualize HTTP server logs, inspired by Atari's pong game, when you get requests, it renders a swarm of pong balls.

In the field of credit fraud, a new method is being employed: the use of graph-based databases. Neo4j [15] and IBM Graph [16], are examples of tools for such purposes. As described in [16] and [15] the motivation is to find cycles inside graphs, which commonly represent a kind of fraud. Neural Networks can also be used as visualization tools. Atienza et al. [17] used Self Organized Maps (SOM) to find web traffic attacks.

Chi [18] surveys a couple of visualization tools developed at the User Interface Research Group at Xerox PARC. The mentioned work used visualization tools to improve web usability, to find and predict browsing patterns and to show web structure and its evolution. Like us [18] also implemented a graph inspired visualization.

Another graph inspired visualization is Hviz [19], which was used successfully by InfoSec Institute to explore and summarize HTTP requests to find common malware like Zeus [20], and also as a tool for forensic analysis [21]. Hviz deserves mention for its versatile use cases and also for creating a heuristic to aggregate HTTP requests by using Frequent Item Mining. Hviz is related to ReSurf [22], using it as a benchmark to improve browsing reconstruction. Another work in this field is ClickMiner [23], which also reconstructs browsing paths and provides a tool to visualize it; this is analogous to the Click Path feature we propose, but the context is different: they analyze traffic from a single machine, client by client, whereas ours is server-based and we don't need access to individual computers.

Blue et al., presented NetGrok [24], which uses a combination of graphs and Treemaps to display bandwidth usage from IP hosts in real-time. Although they used the tool successfully to detect anomalies, the scope of their analysis does not match with ours; they use low-level packet capturing, whereas we use server logs, more close to the Web Analytics resources available.

We propose new ways to display website traffic by using an interactive tool that provides several ways to arrange visits, conversions, user behavior, click path, page views and filtering options. All of them, combined with pattern recognition [9], could help to find clusters for new market niches, discover unknown visitor segments or improve segment analysis by the use of the patterns found on the data. We integrated into the visualization tool a way to introduce a pattern and using it as a query to filter the visits. This allows the expert to create segments automatically (after introducing the pattern) or at least give some insight on which group of visitors shares common properties or behavior.

Most of these reports and graphs are based on user sessions, which are usually identified by cookies, that link a requested resource to a particular visitor. In the case of web log files, cookies' information is not available, so other methods are applied to the discovery of user sessions. Such is the case of the work in [25], which assigns requests to sessions following the next heuristics, in order, (1) same IP addres and user agent, (2) same user agent and common domain (obtained by reverse DNS lookup of the IP address), (3) same user agent and common IP prefix, and 4) same IP address and different user agent. More commonly, session definition consist of joining requests from the same IP address and user agent, as is the case of [26,27]. Additionally, all approaches define a session time out, typically set to 30 min.

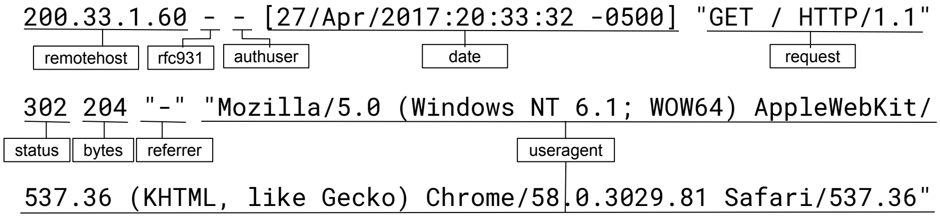
3. Dataset

Our visualization has been designed to work with log files from a web server. Unlike most approaches, we do not use data related to the interaction while browsing a page (mouse usage, supported browser features, etc.), we work solely with data directly available from a standard log file. Using client-side data, logged directly through the client's browser (i.e., by javascript tracking code), may lead to a richer feature space. However, we have found that server data on its own is enough to provide a general idea of the visitors' behaviour on the site. We aim to cover the needs of companies that may be reluctant to add logging scripts due to privacy concerns. A weblog file is a trace already available to servers, and it does not require running any additional script.

We analyzed log data recorded by web servers from a commercial website. In total, we examined the log files of one month of interest. The data was collected by company experts, aiming to provide only navigation requests generated by allegedly human visitors. These logs use an extended version of the NSCA Common Log Format, known as Combined Log Format [28]. Table 1 shows the fields that a web server records in its log files when the Combined Log Format is used. By parsing the log file, we were able to extract these fields, for each line in the file (see Figure 3 for a sample line and its fields).

Table 1. NSCA Common Log Format Fields.

Field	Description
remotehost	IP address
rfc931	The remote logname of the user.
authuser	The username that has been used for authentication.
date	Date and time of the request.
request	Resource requested and HTTP version.
status	The HTTP status code returned to the client.
bytes	The content-length of the document transferred.
referrer	The URL which linked the user to the site.
useragent	The Web browser and platform used by the visitor.



```
200.33.1.60 - - [27/Apr/2017:20:33:32 -0500] "GET / HTTP/1.1"
302 204 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/
537.36 (KHTML, like Gecko) Chrome/58.0.3029.81 Safari/537.36"
```

Figure 3. Example of line in a logfile, using the Combined Log Format.

Next, we performed a series of pre-processing steps in order to obtain an extended representation of each log entry. Given that log files are not intended to be read by the common user, the information (fields) they provide, while valuable, may not be insightful to web marketing teams. For this reason, we extract features from the log fields and create objects that represent log entries. From these objects, it is possible to obtain contrast patterns that describe the characteristics of a group of users, as described in Section 5. In the next paragraphs, we explain the pre-processing steps we followed to extract the feature vector used in our work.

The first field available is the IP address of the visitor. From this field, we can extract geolocation and contextual features. We used GeoLite databases [29] to extract the City, Country, Subdivision and Organization associated with the IP address. Using these geolocation features allows for a more generalized analysis. For instance, using the log field raw values, it is not possible to identify two visitors from the same city but with different IP addresses; whereas in the proposed feature space, they will have a common value. Additionally, the extracted features are more interpretable. An IP address might not tell much to a user; however, knowing the location or the organization of the visitors provides a better idea of their profile.

We skip rfc931 and authuser because they had the same value in all the log entries in our data. Then, we process the date field. As can be seen in Figure 3, the date is logged using the format [dd/MMM/yyyy:hh:mm:ss +-hhmm]. This format is not convenient for data mining because it is very specific and does not allow generalization. Instead, we extracted two features from this string: the hour (rounded up if the time is closest to the next hour) and the day of the week. We do not take minutes into account because they are very specific for our purposes, while hours from 24 different groups, hour and minute precision would allow too many groups to be created. If more precision is desired, instead of taking directly the time including minutes, we recommend to bin times of the day, in order to have more than 24 groups but not as many as 1440. Next, we processed the request to extract the URL and the number of parameters in the URI-query, again, with the purpose of getting a more general feature vector .

We maintained the next three fields (status, bytes, and referrer) as features. Finally, using UAParser [30] we obtained, from the useragent, the operating system, browser and device used by the host. In total, we have a set of 14 features, which are shown in Table 2. These features are used by our pattern mining approach.

Table 2. Feature Set. The first column lists the features that will be used in our analysis. The second column lists the name used to identify the feature in the patterns. The third column specifies the origin of the feature in the logfile.

Feature	Tag	Log Field
Hour	hour	date
Day of the Week	dayOfWeek	date
City	city	IP address
Country	country	IP address
Subdivision	subdivision	IP address
Organization	organization	IP address
URL	url	request
Number of parameters	parameters	request
status	status	status
bytes	bytes	bytes
referrer	referrer	referrer
Operating System	agentOS	useragent
Browser	agentBrowser	useragent
Device	agentDevice	useragent

Additionally, we processed the logs to obtain user visits, which are the main element in our visualization. This processing was made by importing the data to Matomo. Visits include pages requested by the same user (identified by a device fingerprint). When a user requests a page more than 30 min after his last requested page, it is considered a new visit. This allow us to create an extended vector with information regarding the whole visit, instead of one resource request. Table 3 describes extra features that were extracted after processing the data in Matomo. Please note that this list is not an exhaustive list of Matomo's database structure. Features like the number of clicks, plugin flags, and page generation time, among others; are only available when page tracking is performed directly through Matomo tracking code (on the client's side); since we imported server logs into Matomo this data is not available, as it is not possible to infer it from the logs. Additionally, with this processing, we eliminate requests to internal resources with are not required in our current analysis (i.e., javascript code, static image/css information, etc.); alternatively, a list of resources of interest can be obtained from the site owner and used to filter the data.

Table 3. Features from sessions.

Feature	Description
actions	Number of actions in the visit.
daysSinceFirstVisit	Days past since the first time the user visited the site.
daysSinceLastVisit	Days past since the last time the user visited the site.
firstActionTimestamp	Timestamp of the first action in a user's visit
lastActionTimestamp	Timestamp of the last action in a user's visit
pageviews	How many pages the visitor viewed
referrerType	Where the visit comes from. Example: *direct*, *search*, *website*, etc.
timeSpent	Average time the user spent in a page in seconds
visitDuration	Visit duration in seconds
visitorId	ID for identifying unique visitors
visitorType	Can take the values *new* and *returning*.

4. Visual Model

In order to aid the task of understanding website traffic, we have designed an interactive visualization tool that provides visual cues indicating the relationship between visitors and pages, along with analytic metrics. Following the analysis, certain marketing strategies can be proposed to improve the page content or to take advantage of the most visited pages, like including ads into them or adding appealing elements in the pages. The site to be analyzed is not one with millions of visits per day, on Section 6 we specify the design changes we suggest for the analysis of sites with a high volume of visitors.

Specifically, we have two visual elements dedicated for this purpose:

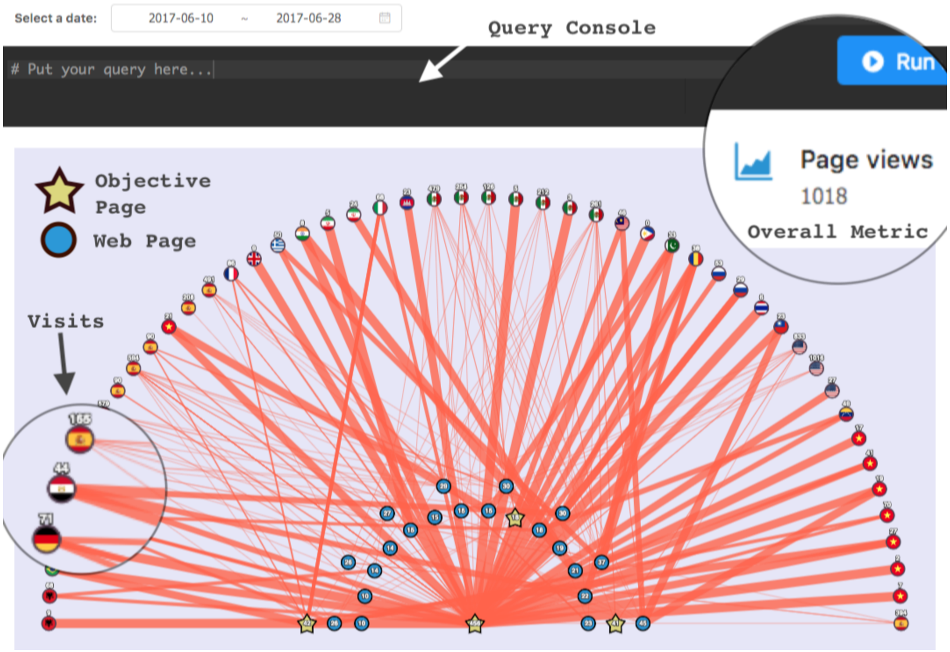
Visits view This visual element allows exploring the relation between pages and visits, while allowing observation of analytic metrics, such as page views, bounce rate, and visit duration. The user can also highlight a single page or visit to observe its relations and metrics.

Navigation path This visual element allows exploring individual visits in a graph structure.

In the following sections, we describe the design of each visual element and present examples of how they aid to the goal of understanding website traffic.

4.1. Visits View

Page and visit reports are common features found in any web analytics platform, often, those reports are provided as tables or line plots (as shown in Figure 1). However, by using a different visualization, we can combine those reports into a single one. Our proposed visualization (Figure 4) displays information about both: the pages in the site and the visitors to these pages. The design is based on several concentric circles, formed by three different type of nodes, which connect to each other according to visit and page relationships. We start by describing the three types of nodes:


Select a date: 2017-06-10 ~ 2017-06-28

Put your query here...

Page views 1018

Figure 4. Visits View. Blue nodes represent web pages; stars are objective pages; nodes with country flags are visits of users from that country. Page views was used as metric and inverse *visit duration* for the edge width. Thin connections represent longer visit duration; thicker otherwise.

Visit nodes Visit nodes form the outermost concentric semi-circle, these nodes are represented as circles with a country flag image. Each node represents a visit to the website. The flag indicates from which country the visit came.

Page nodes Page nodes form several concentric semi-circles, these are the blue circles. Each node represents a unique web page of the website.

Objective nodes Objective nodes are pictured as stars. These nodes are web pages that are considered goals of the business, for example: sign-up pages, landing pages, checkout pages, among others. They are distributed along with the page nodes.

As we can see, nodes are arranged in two groups. The first group contains all the visit nodes and is distributed in the outermost semi-circle. The second group contains all the page and objective nodes and it distributed in several semi-circles. The nodes are ordered within the semicircles. Next, we explain how this ordering was defined, according to visit and page characteristics, and the advantages of this design.

Visit nodes are grouped according to the country of origin. This allows to identify the countries that have more visits, as large groups of nodes with the same flag are easy to spot. An alternative ordering is to sort the nodes according to the visit duration, in this case, the user can look at the nodes at the far right (or left) to observe which visits were the longest (or shortest) and if they had any similarities in their countries of origin. We kept the ordering according to the country of origin due to two reasons: (1) it was preferred by our users, and (2) the visit duration is reflected in the connection between the nodes, as will be explained later in this section.

Page and objective nodes are distributed in k semicircles, also called levels. Each level is ordered from the center to the outermost level in ascending order. Once the parameter k is established, the elements are distributed into the k levels in the following way: k segments of size $\frac{max(metric)}{k}$ are selected, starting from $min(metric)$. Then, we sort all the nodes into the respective level, ordered in ascendant from left to right. The index page, or home page of the website, is excluded from these computations because its position is fixed into the center of the view.

A metric must be selected to designate the order of the nodes. The selection of this metric will determine the sort of information that will be quickly grasped just by looking at the position of the nodes. After continuous iterations with the final users of our tools, the selected metric is page views. Thus, pages with fewer page views are positioned at the leftmost side and increasingly positioned to the right. Pages on the first level (shortest radius) are the ones with fewer page views, whereas pages on the k^{st} level have the highest page views. This allows the marketing team to quickly identify the most visited pages, and see if their starred pages are truly visited more. Additionally, in the center of the node, the metric is displayed.

The selection of k can reflect a certain *Key Performance Indicator (KPI)*. For example, the business could decide to use $k = 5$ and define the goal KPI as having all the objective nodes at the fourth level; if this goal is not achieved it is an indicator of bad performance of the business goals. Ideally, objective nodes should appear in the outermost level.

As an example, we have Figures 5 and 6. Page and objective nodes will be placed in their corresponding level, depending on how many page views they have. In this example, $k = 3$, the most visited page had 45 page views, and the least visited page had only ten views. Thus, we end-up with three segments of size 15 ($SegmentSize = 45/3 = 15$), starting at 10: $[min(pageviews), 25]$, $(25, 41]$, and $(41, max(pageviews)]$. In this case, we have 14 pages in the first level, 8 in the second level, and two in the third level. Figure 5 shows a scenario where the objective nodes are most viewed pages, this indicates an ideal business scenario. On the contrary, Figure 6 shows a scenario where the objective nodes lay on the inner-most semicircle, i.e., they are the within the less viewed pages. This indicates a bad business scenario. Nevertheless, our visualization is useful to spot new opportunities by looking at page nodes in last semi-circle. There pages are currently not considered as objectives, but they have a lot of views. Thus, we can implement some call-to-action elements inside there, such as advertising, banners, promotions, among others.

The final element of this visual component are the connections between the nodes. A connection between a visit node and a page or objective node indicates that such page was accessed by that visitor. The width of the connection represents a selected metric, in this case, we chose the inverse of the visit duration. The less time spent visiting the website, the thicker the connection. On top of the visit nodes, the metric used for the width of the connection is displayed; in this case, how many seconds the visit lasted.

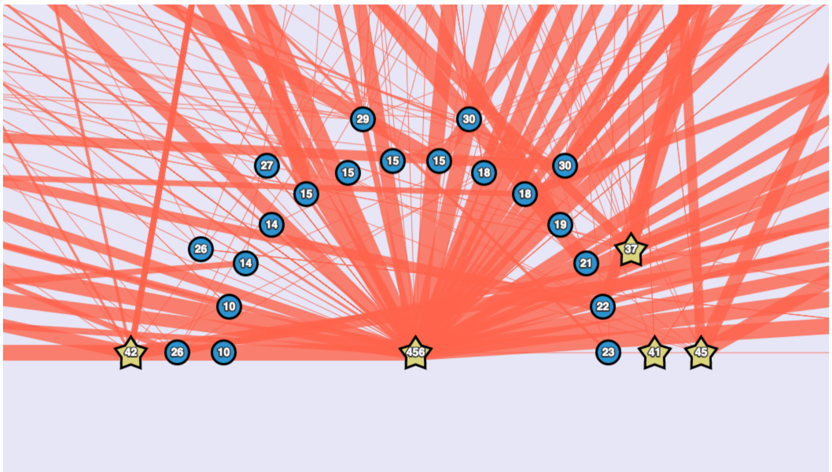


Figure 5. Performance visualization through a concentric layout: ideal scenario.

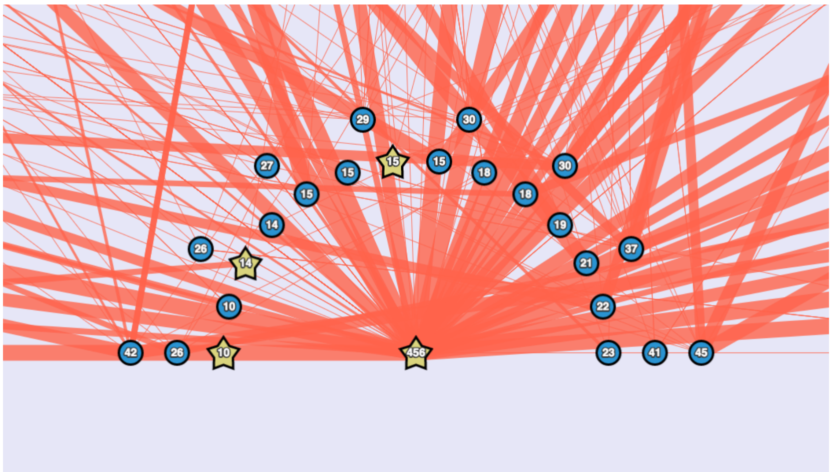
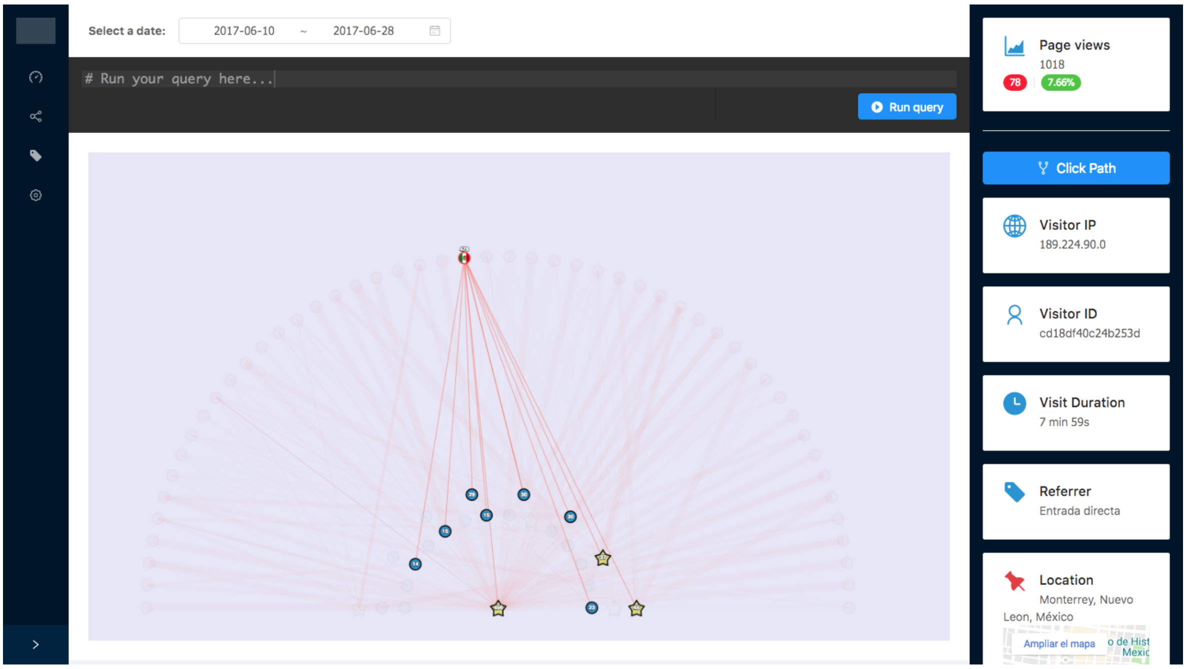


Figure 6. Performance visualization through a concentric layout: bad scenario.

This visual component, not only displays information, but it also is interactive. The use can click on nodes to obtain relevant information about the node. When selecting a visit node, the connections and nodes not related to this visit are dimmed and the sidebar is populated with information about the selected visit, including a button that allows access to the Navigation Path (Section 4.2). This way, the user sees all the pages related to the selected visit and information such as visit duration and location. This behaviour is shown in Figure 7. When selecting a page or objective node, the connections and nodes not related to this page are dimmed and the sidebar is populated with information about the selected page, including the url, number of pageviews and average time on the page. This behaviour can be seen in Figure 8. If a star is seen in the inner levels of the semi-circle, the user can click on it, to show the details of the page and device new strategies to bring visitors to this page. Likewise, a user can identify a page previously thought as not important, to be, in fact, one of the most visited; in this case, the team can add to this page information that they want visitors to see.

We have shown that by using a single visual report, we can quickly observe page views, with an easy access to detail, as desired by the marketing team, including the performance of goal pages, plus the country from which the visit comes from. Additionally, instead of having text indicators, users can quickly glance at the relationship of visitors and pages, including the visit duration. The selection of these metrics was the result of a continuous feedback process with the marketing team of the analyzed e-commerce site. Choosing page views allows an easy identification of the most popular pages. Choosing the visit duration allows an easy identification of visitors who spend more time in the page and are probably interested in the site (their navigation path is interesting to analyze) whereas visitors with a short visit duration might correspond to bots or uninterested visitors.



Select a date: 2017-06-10 ~ 2017-06-28

```
# Run your query here...
```

Page views
1018
78 7.66%

Visitor IP
189.224.90.0

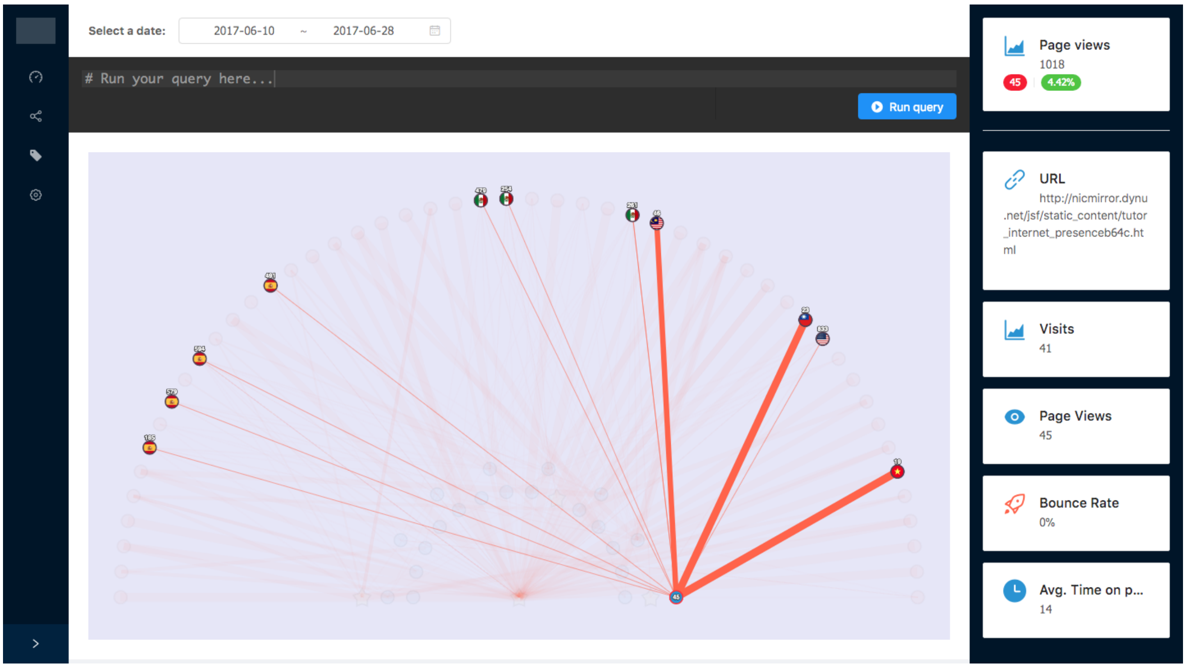
Visitor ID
cd18df40c24b253d

Visit Duration
7 min 59s

Referrer
Entrada directa

Location
Monterrey, Nuevo Leon, México

Figure 7. Visits view: selecting a visit.



Select a date: 2017-06-10 ~ 2017-06-28

```
# Run your query here...
```

Page views
1018
45 4.42%

URL
http://nicmirror.dynu.net/jsf/static_content/tutor_internet_presenceb64c.html

Visits
41

Page Views
45

Bounce Rate
0%

Avg. Time on p...
14

Figure 8. Visits view: selecting a page.

4.2. Navigation Path (F4)

In many platforms, the user navigation path, is commonly represented as tables of sequential actions or as aggregated traffic flow (Figure 2). We believe this could be improved using a visual representation. In this section, we describe our proposed visual representation, inspired in network diagrams.

Figure 9 shows the proposed visualization for the navigation path. It includes a series of page nodes and objective nodes connected when the visitor navigated between the corresponding pages. Each node represents a page the user visited, as before, blue circles represent common pages whereas stars represent objective pages. The user can see the Universal Resource Identifier (URI) of each node.

Additionally, each connection has two items of information. The first element shows the order in the chain of viewed pages and the second element is the time spent by the user in the page.

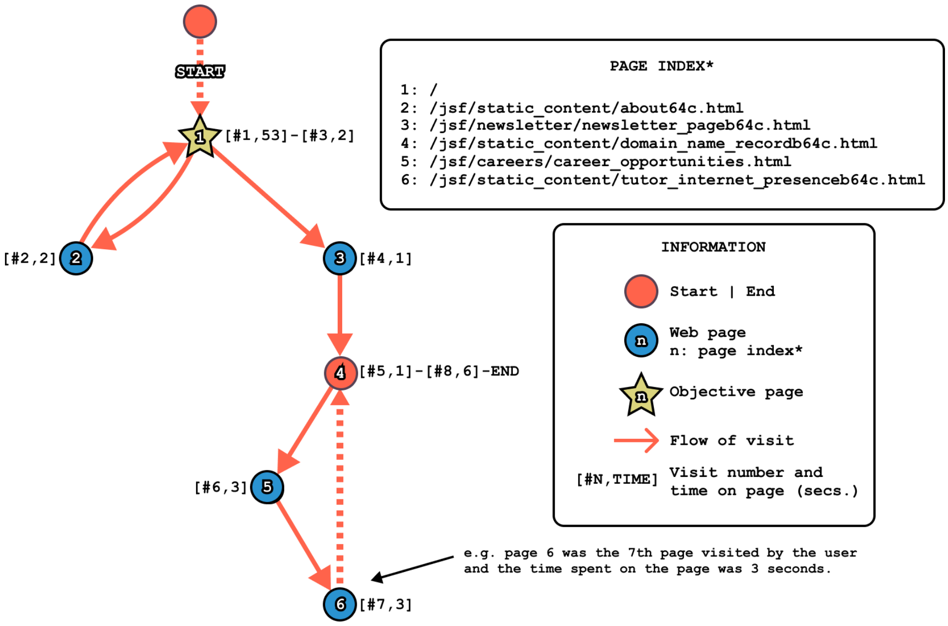

START
1
[#1,53]-[#3,2]
[#2,2]
2
3
[#4,1]
4
[#5,1]-[#8,6]-END
[#6,3]
5
6
[#7,3]
PAGE INDEX*
1: /
2: /jsf/static_content/about64c.html
3: /jsf/newsletter/newsletter_pageb64c.html
4: /jsf/static_content/domain_name_recordb64c.html
5: /jsf/careers/career_opportunities.html
6: /jsf/static_content/tutor_internet_presenceb64c.html
INFORMATION
Start | End
n
Web page
n: page index*
n
Objective page
Flow of visit
[#N,TIME]
Visit number and
time on page (secs.)
e.g. page 6 was the 7th page visited by the user
and the time spent on the page was 3 seconds.

Figure 9. Click Path. Navigation graph of a visit. The number at the edge's center indicates the view step, and the one with a clock icon shows how long the previous page was viewed.

Our visualization, unlike that presented in Figure 2, gives a quick understanding of non-linear navigation. For example, the presence of loops and recurrent pages is now straightforward to appreciate. Additionally, our click path visualization, enables identifying at which point of the visit the visitor landed to a certain objective page, or if in fact, the visitor never touched an objective. However, a limitation of obtaining this path directly from log files happens when the user clicks the browser back button, as there will be no entry in the log for this event.

5. Traffic Segmentation and Characterization

One of the first tasks for analyzing an audience is to create segments, which involves performing a filter process given a series of conditions: e.g., young people ($18 < \text{age} < 24$); men living in New York City (gender = male AND location = NYC); people coming from social networks (referrer IN [Facebook, Twitter, Snapchat]), etc. So, it is very important to have an easy way to perform such filters. In this section we describe how filtering is integrated into our visualization and how a pattern mining approach can aid the characterization of segments. Finally, we present two case study scenarios where we applied our approach.

5.1. Filtering Visits View

We incorporated a query console to allow users to select a group of nodes that meet a certain criteria (notice the console at the top in Figure 4). The query console is an interface to the open-source project called Cytoscape by [31], which is responsible for the actual query expression evaluation. The queries that can be performed using the tool, supporting the base logic operators: $\wedge, \vee, \neg$ (AND, OR, NOT), relational operators ($=, >, <, \leq, \geq$), string matching, and others. The patterns follow a very similar syntax than the Disjunctive Normal Form (DNF, [32]). All capabilities are provided by Cytoscape and the full specification can be found in the documentation [31].

For the visual model, we import the log files to Matomo, so we are able to query visits and pages based on their database structure [33]. The format follows the next grammar:

$$group[attr\ OP\ val], group \in \{\texttt{node}, \texttt{edge}, .className\}$$

where:

- *className* represents a custom class assigned to a particular data. In our case we have three classes: `visit`, `page`, and `objective`. So instead of using `node` or `edge`, we use such classes which are less abstract. For example, `.page`$[attr\ OP\ val]$ will target only web page attributes, similarly `.visit`$[attr\ OP\ val]$ will query for the visit information.
- *attr* represents the attribute used for filtering. Such attributes correspond to the context represented by the group. For the `visit` group, the attributes are: `duration`, `country`, `browser`, `events`, among others. In the case of the `page` group the attributes are: `url`, `visits`, `pageviews`, `bounceRate`, `exitRate`, `avgTimeSpent`, among others.
- *OP* represents any binary operator. Depending on the data type, certain operators can be used over the others. The available operators are: `=, !=, >, <, >=, <=, *=,` $\wedge$`=, $=`.
- *val* is the value used for matching *attr* by the selected operator (*OP*). Depending on the data type of *attr*, *val* is a string, a number or a boolean.

The logical operators $\wedge$, and $\vee$ use the following format:

$\wedge$ (**AND**):
$group_1[attr_1\ OP_1\ val_1]group_2[attr_2\ OP_2\ val_2] \ldots\ group_n[attr_n\ OP_n\ val_n]$
Notice the join of groups after the square brackets.

$\vee$ **(OR)**:
$group_1[attr_1\ OP_1\ val_1], group_2[attr_2\ OP_2\ val_2], \ldots, group_n[attr_n\ OP_n\ val_n]$
Notice the use of the *comma (,)* for concatenating conditions.

The previous querying system enables us to enter patterns obtained with machine learning algorithms, such as the one presented in the next Section. In Table 4, we provide examples of how to use such query system to select a segment of traffic. In addition, in Figure 10 we can observe an example of the Visits view with a filter applied, highlighting the group of nodes that meet the specified criteria.

Table 4. Query examples to filter visualization visit views.

Use Case	**Segment Description**	**Query**
Simple query with the .visit className	Visitors from the United States	`.visit[countryCode = ‘us’]`
Using the AND operator	Visitors using Google Chrome AND with a visit duration greater than 10 s AND coming Mexico	`.visit[browserCode = ‘CH’] [visitDuration>10] [countryCode = ‘us’]`
Using the OR operator	Visitors coming from Facebook OR Instagram	`.visit[referrerName = ‘Facebook’], .visit[referrerName = ‘Instagram’]`
Using both AND and OR operators	Visitors from Mexico OR visitors from the United States that have visited the site more than once	`.visit[visitCount>1].visit[countryCode = ‘us’], .visit[countryCode = ‘mx’]`

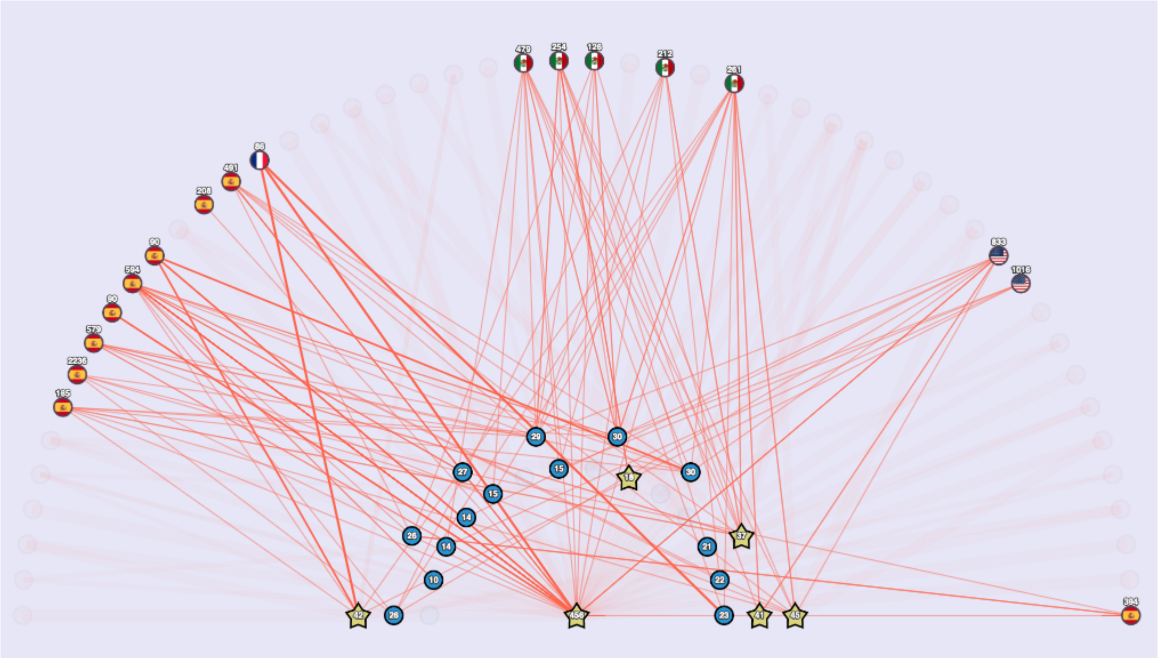


Figure 10. Result of running a query. Only the complying nodes are highlighted.

5.2. Pattern Mining Algorithm

A pattern mining approach can aid traffic segmentation in two different aspects: by helping to discover interesting segments, and by characterizing such segments. Contrast pattern-based classifiers are an important family of both understandable and accurate classifiers [9]. A pattern is an expression defined in a certain language that describes a collection of objects. For example, $[visit\_duration \leq 5] \wedge [country = \text{“}China\text{”}] \wedge [number\_of\_click \in [0,5]]$ is a pattern describing a bot behavior.

A contrast pattern is a pattern appearing significantly in a class regarding to the other classes [9,34–36]. These patterns describe common characteristics of objects in a class. We propose to define the classes based on the segments of interest. Contrast pattern-based classifiers have been used on several real-world applications like characterization for subtypes of leukemia, classification of spatial and image data, structural alerts for computational toxicology and prediction of heart diseases; among others, where they have shown good classification results [9,36–40].

Mining contrast patterns is a challenging problem because of the high computational cost due to the exponential number of candidate patterns [9,41]. Also, some algorithms for mining contrast patterns need an *a priori* global discretization of the features, which might cause information loss. For this reason, those pattern mining-based approaches avoiding a global discretization step, allowing low computational cost, and obtaining a small collection of high-quality patterns, have special attention for the international community; an example of this are those contrast pattern miners based on decision trees [41].

Usually, contrast pattern miners based on decision trees build several decision trees for extracting several patterns from them. For each decision tree, patterns are extracted from the paths from the root node to the leaves. For each extracted pattern, the class with the highest support determines the class of the pattern [36,40,41].

From those contrast pattern miners based on decision trees, Random Forest miner have proved better classification results and better diversity of high-quality patterns than other approaches based on decision trees for mining contrast patterns [41]. Also, Random Forest miner have allowed obtaining better classification results when Hellinger distance [42] is used for evaluating each binary candidate split at each decision tree level than using the information gain measure [43]. This result is due to the

Hellinger distance is unaffected by the class imbalance problem because it rewards, in a better way than the information gain measure [36].

For all of the above reasons, in this paper, we have selected the Random Forest miner jointly with the Hellinger distance as the algorithm for mining contrast patterns. Also, we used the pattern filtering method introduced in [36] for obtaining a small collection of high-quality pattern describing the problem's classes, in this case the segments of interest. These patterns can guide the analysis of the data, by pointing to interesting visitor patterns; guiding the user towards creating queries to filter the data, instead of guessing interesting subsets of visitors that may appear in the data. Additionally, we have noticed that patterns obtained contrasting two classes can point to other segments that could be interesting to analysts. In the next section, we present a case study where we observe this phenomenon.

5.3. Case Study Scenario: Humans vs. Bots

Often, the first step of visitor analysis on web sites is to perform a data cleaning process to eliminate bot visits from the analysis. We present the bot versus human analysis, as an example of how pattern mining can be applied to characterize two visitor segments. We used data mining techniques to extract patterns which marketing teams can observe to get interesting trends from both types of visitors.

We used the dataset described in Section 3. For this scenario, the pattern mining algorithm will work with two classes: human and bot; which were obtained using a one-class classifier (BaggingRandomMiner [44]). Next, we will show the extracted patterns separating human from bot traffic from a specific day of interest. Note that the analysis does not have to cover only a particular day, it can cover the desired period of interest. For example, if a marketing campaign lasted four days, the patterns can be extracted from that time span. Alternatively, it may be of interest to feed the system with visits from certain hours; for example, as will be seen shortly, patterns may outline an interesting subset of visits, which is of interest to analysts.

Patterns with high support for one class are of interest because they show a general behavior of that class. Also, patterns without necessarily high support but with zero support for the other class are also interesting because they show subgroups of visitors. First, we have a pair of related patterns that characterize the normal class:

(A) $\text{country} = \text{``}\textit{Mexico}\text{''} \wedge \text{hour} \leq 10 \wedge \text{agentOS} \neq \text{``}\textit{Other}\text{''}$
(B) $\text{country} = \text{``}\textit{Mexico}\text{''} \wedge \text{hour} > 10$

Both patterns have a support of zero for the anomalous class, which means that they do not cover bot visitors. Additionally, pattern A has a support of 0.1107 and pattern B of 0.8574. Given the nature of these particular patterns (the intersection of the visitors covered by both patterns is zero), together, they account for 96.81% of the human visitors. We can conclude that for this day, at least 96% of the human visitors come from Mexico. Many more visits (85.74%) are registered after 10 a.m. than those registered before this time (11.07%). This can serve as an indicator that only a few users visit the page early in the morning, most users visit after 10 a.m. Another pattern with zero support for the bot class and high support for the human class is:

(C) $\text{country} = \text{``}\textit{Mexico}\text{''} \wedge \text{referer} \neq \text{``?''}$

Pattern C has a support of 0.9618 for the normal class. This pattern states a high percentage of the users (96.18%) come from Mexico, as we already know by pattern A and B; but it is interesting because it adds the information that these visitors arrive through a referrer.

We found that our one-class classifier was able to find visits from two known crawlers, even when the provided log files were pre-selected as human behavior. Next, we show some patterns with zero support for the human class, which means that only bots follow each of them:

(D) $\text{agentBrowser} = \text{``}\textit{Sogouwebspider}\text{''}$

(E) country $\neq$ "*Mexico*" $\wedge$ agentOS $\neq$ "*Other*" $\wedge$ agentBrowser = "*BingPreview*"
(F) city = "*Redmond*"
(G) city $\neq$ "*Redmond*" $\wedge$ subdivision = "*Beijing*" $\wedge$ *url* $\neq$ "*robots.txt*"
(H) city = "*Sunnyvale*" $\wedge$ subdivision $\neq$ "*Beijing*" $\wedge$ referer $\neq$ "?"

In the analyzed day, Sogou web spider (Chinese search engine crawler) and Bing Preview (used to generate page snapshots) appeared in patterns D and E, with a support of 0.2372, and 0.2146, respectively. Other patterns describing bot visitors, like F, G, and H, were inclined to geolocation features, indicating us a fraction of the bots come from Beijing (pattern G support: 0.2514), Redmond (pattern F support: 0.2571), and Sunnyvale (pattern H support: 0.1356).

In this case, we took the class label provided by the one-class classifier to extract the patterns. Alternatively, we can extract patterns using a different segmentation. Furthermore, the patterns obtained can help to point towards a new segmentation. For example, pattern C states that most human users in the time of interest arrived through a referrer, this may motivate to characterize users based on their referrer. Another interesting analysis, suggested by patterns A and B, would be to characterize traffic from before and after ten.

5.4. Case Study Scenario: Segmentation by Country

Next, we applied the proposed approach to contrast visitors from Mexico versus those from Asia. Again, we aim to extract patterns which marketing teams can observe to get interesting trends from both types of visitors.

For this scenario, the pattern mining algorithm will work with two classes: Mexico and Asia. The first step is to label the data. We obtain class label by manually labelling the data depending to the country associated with each visit. If a visit does not belong to either class, it is left out of the analysis. Next, we will present the results in the form of a summary of distinctive traits.

Again, we analyzed the patterns which have zero support for the opposite class, given that they show characteristics exclusive to the class of interest. In the Mexican class, we observe two interesting patterns related to the visit duration. These patterns tell us that half of visits from Mexico have a duration longer than 100 s and 13 percent have a duration of less than five seconds; from this we can infer that the rest of the visitors (37 percent) have a visit duration between 5 and 100 s. A possible business goal could be to try to engage such 13% of visitors that stay in the site less than five seconds.

Also, since the patterns have zero support, we can further and infer that Asian visitors browse the site an stay at most 100 s. Again, this can be actionable information, if the business is interested expanding to a new market, then it can aim to optimize the duration of Asian visits.

6. Further Work

New visualization models, complemented with a pattern mining approach, are useful to discover common characteristics of users and to understand their behaviour on a site. Here we presented an approach aimed at companies which do not yet have thousands or millions of visitors per day. Here, we presented two case study scenarios to showcase the capabilities of pattern mining applied to segmentation of web log data. In the analyzed month the site reaches at most hundreds of visits per day, we are currently solving how will the model be adapted, as the number of visits starts increasing. In the next paragraphs, we describe opportunities to extend our approach, and how they would be appealing to the analysts of a more popular site.

The first challenge when the traffic volume increases is the number of nodes visible in our visual element Visit views; either it the visualization start to saturate, or only a sample of the nodes are displayed. To amend this, we suggest joining similar nodes into a bigger node, which can then be expanded if this group of nodes is of interest to the analyst. To decide what qualifies as a similar node, a metric or feature can be used. We believe that geographical (for example by country, city, or region) can be useful to evaluate the reach of a particular marketing campaign.

Additionally, our prototype allows the user to observe the navigation paths of individual users. This feature was particularly well received by the marketing team with whom we collaborated. Other tools do not display such a graph for individual analysis, instead, their visual models show information for general analysis. In this scenario, patterns can help to select which, out of the many website visits, may be of interest to observe individually. The navigation path of a visitor, allows the analyst to validate if visitors really navigate the way they expect and helps them to notice, for example, if there is a page that has the attention of the users (loops create around a particular page node) or one where they leave the page (last page node in the graph).

There are many opportunities to extend our tool. For example, the automatic analysis of navigation graphs could provide marketing teams with models or patterns that define the behavior of their visitors. Graphs of all visitors could be seen as sequences and mined to look for interesting insights such as the most frequent sequences, the most frequent landing page (first element of the sequence), etc. This is especially useful when the traffic volume increased, with graph analysis the repetitive navigation patterns can be found and, thus, there is no need to look at individual patterns which would make the task overwhelming. Additionally, graphs could also be analyzed to detect subgroups of users, for example, or bot patterns, given that bots probably have a more structured or linear way of visiting the pages on a site whereas humans navigation is probably more complex structure and contains loops. An analysis of navigation graphs can also be useful to identify sequences that lead to conversions, which can be useful to reconfigure the site.

Our prototype is meant to be a complement to, not a replacement of, web analytic tools, such as Google Analytics and Matomo. Independently of traffic volume, it can be personalized with different metrics for node ordering and connection width between the nodes. Here we presented metrics tailored to our final users interest. As is, it already provides interesting features, we believe the combination of machine learning and visualization techniques is a promising area.

Author Contributions: Conceptualization, B.C., F.G., O.L.-G., M.A.M.-P., R.M. and J.R.-M.; Data curation, B.C., F.G. and O.L.-G.; Formal analysis, F.G., O.L.-G., R.M. and J.R.-M.; Funding acquisition, O.L.-G., M.A.M.-P., R.M. and J.R.-M.; Investigation, B.C., F.G., O.L.-G., R.M. and J.R.-M.; Methodology, B.C., F.G., O.L.-G., M.A.M.-P., R.M. and J.R.-M.; Project administration, O.L.-G., R.M. and J.R.-M.; Resources, O.L.-G., R.M. and J.R.-M.; Software, B.C., F.G., O.L.-G. and M.A.M.-P.; Supervision, O.L.-G., M.A.M.-P., R.M. and J.R.-M.; Validation, B.C., F.G. and O.L.-G.; Visualization, B.C., F.G., O.L.-G., R.M. and J.R.-M.; Writing—original draft, B.C., F.G. and O.L.-G.; Writing—review & editing, B.C., F.G., O.L.-G., M.A.M.-P., R.M. and J.R.-M.

Funding: This work was supported in part by Network Information Center México. Fernando Gómez was partly supported by Consejo Nacional de Ciencia y Tecnología scholarship 470302.

Acknowledgments: We thank the members of the GIEE-ML group at Tecnologico de Monterrey and the members of NIC Mexico for the valuable discussions in this research. We also thank NIC México for providing the web log files used in the experimentation.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Ad Words. Available online: https://adwords.google.com/home/ (accessed on 9 June 2018).
2. Double Click. Available online: https://www.doubleclickbygoogle.com/ (accessed on 9 June 2018).
3. ComScore. comScore: Measure What Matters to Make Cross-Platform Audiences and Advertising More Valuable. Available online: https://www.comscore.com (accessed on 18 June 2018).
4. comScore. Invalid Traffic. 2016. Available online: http://www.comscore.com/Products/Advertising-Analytics/Invalid-Traffic (accessed on 18 June 2018).
5. Brian Pugh. Battling Bots: comScore's Ongoing Efforts to Detect and Remove Non-Human Traffic. 2012. Available online: https://www.comscore.com/esl/Insights/Blog/Battling-Bots-comScores-Ongoing-Efforts-to-Detect-and-Remove-Non-Human-Traffic (accessed on 18 June 2018).
6. KissMetrics. Kiss Metrics Platform. 2017. Available online: https://www.kissmetrics.com (accessed on 9 June 2018).

7. Matomo. Matomo. Available online: https://matomo.org/ (accessed on 12 October 2018).
8. Peter Adams. Open Web Analytics Repository. Available online: http://www.openwebanalytics.com/https://github.com/padams/Open-Web-Analytics (accessed on 18 June 2018).
9. Dong, G.; Bailey, J. *Contrast Data Mining: Concepts, Algorithms, and Applications*; CRC Press: Boca Raton, FL, USA, 2012.
10. García-Borroto, M.; Martínez-Trinidad, J.F.; Carrasco-Ochoa, J.A.; Medina-Pérez, M.A.; Ruiz-Shulcloper, J. LCMine: An efficient algorithm for mining discriminative regularities and its application in supervised classification. *Pattern Recognit.* **2010**, *43*, 3025–3034. [CrossRef]
11. Gutierrez-Rodríguez, A.E.; Martínez-Trinidad, J.F.; García-Borroto, M.; Carrasco-Ochoa, J.A. Mining patterns for clustering using unsupervised decision trees. *Intell. Data Anal.* **2015**, *19*, 1297–1310. [CrossRef]
12. Akamai. Real-Time Internet Monitor Akamai. Available online: https://www.akamai.com/us/en/solutions/intelligent-platform/visualizing-akamai/real-time-web-monitor.jsp (accessed on 26 June 2018).
13. Kaspersky. Kaspersky Cyberthreat Real-Time Map. Available online: https://cybermap.kaspersky.com/ (accessed on 26 June 2018).
14. Logstalgia. Logstalgia—A Website Access Log Visualization Tool. Available online: http://logstalgia.io/ (accessed on 28 June 2018).
15. Neo4j. White Paper: Fraud Detection Discovering Connections—Neo4j Graph Databas. 2015. Available online: https://neo4j.com/resources/fraud-detection-white-paper/ (accessed on 9 September 2017).
16. Mahmoud, A. Detecting Complex Fraud in Real Time with Graph Databases—The Developer Works Blog. 2017. Available online: https://developer.ibm.com/dwblog/2017/detecting-complex-fraud-real-time-graph-databases/ (accessed on 9 September 2017).
17. Atienza, D.; Herrero, Á.; Corchado, E. Neural analysis of HTTP traffic for web attack detection. *Adv. Intell. Syst. Comput.* **2015**, *369*, 201–212. [CrossRef]
18. Chi, E.H. Improving web usability through visualization. *IEEE Internet Comput.* **2002**, *6*, 64–71. [CrossRef]
19. Gugelmann, D.; Gasser, F.; Ager, B.; Lenders, V. Hviz: HTTP(S) traffic aggregation and visualization for network forensics. *Digit. Investig.* **2015**, *12*, S1–S11. [CrossRef]
20. Institute, I. Botnets Unearthed—The ZEUS BOT. 2013. Available online: http://resources.infosecinstitute.com/botnets-unearthed-the-zeus-bot/ (accessed on 21 February 2018).
21. DFRWS. DFRWS 2009 Forensics Challenge Challenge Data and Submission Details. Available online: http://old.dfrws.org/2009/challenge/submission.shtml (accessed on 21 February 2018).
22. Xie, G.; Iliofotou, M.; Karagiannis, T.; Faloutsos, M.; Jin, Y. Resurf: Reconstructing web-surfing activity from network traffic. In Proceedings of the IFIP Networking Conference, Brooklyn, NY, USA , 22–24 May 2013; pp. 1–9.
23. Neasbitt, C.; Perdisci, R.; Li, K.; Nelms, T. ClickMiner: Towards Forensic Reconstruction of User-Browser Interactions from Network Traces. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, 3–7 November 2014; pp. 1244–1255. [CrossRef]
24. Blue, R.; Dunne, C.; Fuchs, A.; King, K.; Schulman, A. Visualizing real-time network resource usage. *Vis. Comput. Secur.* **2008**, *5210*, 119–135. [CrossRef]
25. Tan, P.N.; Kumar, V. Discovery of Web Robot Sessions Based on Their Navigational Patterns. *Data Min. Knowl. Discov.* **2002**, *6*, 9–35. [CrossRef]
26. Stevanovic, D.; An, A.; Vlajic, N. Feature evaluation for web crawler detection with data mining techniques. *Expert Syst. Appl.* **2012**, *39*, 8707–8717. [CrossRef]
27. Suchacka, G. Analysis of aggregated bot and human traffic on e-commerce site. In Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, 7–10 September 2014; pp. 1123–1130. [CrossRef]
28. Foundation, T.A.S. *Log Files—Apache HTTP Server Version 2.5*; Technical Report; The Apache Software Foundation. Available online: https://httpd.apache.org/docs/trunk/logs.html (accessed on 16 May 2018)
29. MaxMind's GeoLite2 Dataset. Available online: https://dev.maxmind.com/geoip/geoip2/geolite2/ (accessed on 23 May 2018).
30. Enemærke, S.; Aziz, A. UAParser, C# library. Available online: https://github.com/ua-parser/uap-csharp (accessed on 23 May 2018).
31. Franz, M.; Lopes, C.T.; Huck, G.; Dong, Y.; Sumer, O.; Bader, G.D. Cytoscape.js: A graph theory library for visualization and analysis. *Bioinformatics* **2015**, *32*, 309–311. [CrossRef] [PubMed]

32. Ben-Ari, M. *Mathematical Logic for Computer Science*; Springer: Berlin/Heidelberg, Germany, 2012.
33. Matomo Database Schema. Available online: https://developer.piwik.org/guides/persistence-and-the-mysql-backend (accessed on 26 July 2018).
34. Loyola-González, O.; García-Borroto, M.; Martínez-Trinidad, J.F.; Carrasco-Ochoa, J.A. An Empirical Comparison Among Quality Measures for Pattern Based Classifiers. *Intell. Data Anal.* **2014**, *18*, S5–S17. [CrossRef]
35. Loyola-González, O.; Martínez-Trinidad, J.F.; Carrasco-Ochoa, J.A.; García-Borroto, M. Effect of class imbalance on quality measures for contrast patterns: An experimental study. *Inf. Sci.* **2016**, *374*, 179–192. [CrossRef]
36. Loyola-González, O.; Medina-Pérez, M.A.; Martínez-Trinidad, J.F.; Carrasco-Ochoa, J.A.; Monroy, R.; García-Borroto, M. PBC4cip: A new contrast pattern-based classifier for class imbalance problems. *Knowl.-Based Syst.* **2017**, *115*, 100–109. [CrossRef]
37. Martínez-Díaz, Y.; Hernández, N.; Biscay, R.J.; Chang, L.; Méndez-Vázquez, H.; Sucar, L.E. On Fisher vector encoding of binary features for video face recognition. *J. Vis. Commun. Image Represent.* **2018**, *51*, 155 – 161. [CrossRef]
38. Martínez-Díaz, Y.; Méndez-Vázquez, H.; López-Avila, L.; Chang, L.; Sucar, L.E.; Tistarelli, M. Toward More Realistic Face Recognition Evaluation Protocols for the YouTube Faces Database. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 526–5268. [CrossRef]
39. González-Soler, L.J.; Chang, L.; Hernández-Palancar, J.; Pérez-Suárez, A.; Gomez-Barrero, M. Fingerprint Presentation Attack Detection Method Based on a Bag-of-Words Approach. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*; Mendoza, M., Velastín, S., Eds.; Springer International Publishing: Cham, Swtherland 2018; pp. 263–271.
40. Loyola-González, O.; Medina-Pérez, M.A.; Hernández-Tamayo, D.; Monroy, R.; Carrasco-Ochoa, J.A.; García-Borroto, M. A Pattern-Based Approach for Detecting Pneumatic Failures on Temporary Immersion Bioreactors. *Sensors* **2019**, *19*, 414. [CrossRef] [PubMed]
41. García-Borroto, M.; Martínez-Trinidad, J.F.; Carrasco-Ochoa, J.A. Finding the best diversity generation procedures for mining contrast patterns. *Expert Syst. Appl.* **2015**, *42*, 4859–4866. [CrossRef]
42. Cieslak, D.; Hoens, T.; Chawla, N.; Kegelmeyer, W. Hellinger distance decision trees are robust and skew-insensitive. *Data Min. Knowl. Discov.* **2012**, *24*, 136–158. [CrossRef]
43. Quinlan, J.R. *C4.5: Programs for Machine Learning*; Morgan Kaufmann Publishers Inc.: San Mateo, CA, USA, 1993; p. 302.
44. Camiña, J.B.; Medina-Pérez, M.A.; Monroy, R.; Loyola-González, O.; Villanueva, L.A.P.; Gurrola, L.C.G. Bagging-RandomMiner: A one-class classifier for file access-based masquerade detection. *Mach. Vis. Appl.* **2018**. [CrossRef]