Reporting

SLQ Wiki Fabrication Lab 2024/09/27 13:42

Reporting

WikiRT - SLQ Wiki Reporting Tool

WikiRT is a cross-platform, open source, application developed in-house to analyse, filter and clean up the wiki's access log file and produce accurate data for reporting activity on SLQ Wiki.

Background

Google Analytics is the State Library's primary method of tracking site visitation. When applied to the SLQ Wiki subdomain, page views and sessions are not accurately reported [1)], and there is no capability to report on downloads as per data dictionary requirements. This issue has been discussed with marketing and web services, with a solution developed by the Applied Creativity team, which was approved, then used to report quarterly numbers. However, the data gathering process was manual, and resulted in unacceptable delays in reporting, so an automated process was developed.

How it works

Step 0 – Converting the Apache log to a Dataframe

Converts log to Pandas Dataframe by specifying separating characters, filtering columns, creating table headers. In addition, converts the date and time into a workable format so that it is automatically recognised as a timestamp and can be filtered and sorted as needed using Pandas.

TIME	Date and time of request (includes time zone by default)
IP ADDRESS	Identifies the visitor
USERNAME	Visitor username (only works if visitor has signed in)
REFERER	What server the request was referred from
REQUEST	The wiki page or resource that was requested
USER AGENT	Browser (and version), Operating system and hardware type used to send request

Step 1 – Date masking

Drops dates outside desired timeframe. The from and to dates are specified in the UI by the user.

Step 2 - Filtering

Filters out any known bots and web crawlers. This list can be updated and added to if there are any more identified. the bots and crawlers have been identified using both manual and automatic methods. These are based on keywords and common SQL injection queries, affecting REFERER,

USER_AGENT and REQUEST (SQL queries only).

An example of a crawler that our initial keywords didn't pick up was WordPress jetmon which every 7 minutes (for the duration of the log) requested the exact same wiki page from the same IP address, using the same REFERER and USER AGENT for a total of approximately 10 000 queries. This is quite clearly an automated request and will therefore be filtered.

There are currently two lists; one for bots and crawlers and one for SQL queries (automated attempt to try to access database information via something called SQL injections).

Step 3 - Sessions

It was advised to only count visits from a user once per session, currently set to 30 mins. Timestamps are rounded down to the frequency before dropping duplicates where both the IP and Time match, only keeping the first occurrence.

Step 4 – Removing rows without user agent information

It is highly unlikely that a legitimate request doesn't have a USER AGENT. Every modern browser, will send the user agent information. If the server logs a request without this information it either means someone is trying to play the system or there was an error somewhere in the request. as a result, drop all rows without user agent.

Step 5 – Calculating file downloads

Counts requests for specified file types (images not included).

Step 6 – Calculating unique visits

Outputs the number of rows remaining in the file after the processing Step 1-5.

Step 7 – Exporting csv with results

Remaining data is exported to a new CSV file.

WikiRT app features:

- Analyze raw Apache access log
- Filter out unwanted keywords such as requests made by bots and web crawlers
- Removes entries that are missing crucial information
- Splits user activity into sessions (30mins currently as to reflect SLQ website analytics)

- Outputs clean csv file with results for further analysis

Development tools

This script/application is written in the python programming language and uses the following open-source libraries:

- Pandas is a fast, powerful, flexible and open source data analysis and manipulation tool
- PyQT5 allows the creation of cross platform graphical user interfaces
- Helper libraries: numpy, datetime, pytz

TODO

- Unordered List Item

1)
based on a manual comparison of server access logs (filtering for bots/SLQ injection attacks/crawlers/etc) and GA page views. Primary cause is client-side disabled first-party cookies and/or javascript by use of adblocking browser extensions