Universiteit Leiden

Master Computer Science

Web Tracking in the Netherlands: Insights Into a Comprehensive Database

Name: Dániel Loránt Gelencsér
Student ID: 3228479

Date: 17/11/2025

Specialisation:
Advanced Computing and Systems

1st supervisor: Dr. Akrati Saxena
2nd supervisor: Dr. Saber Salehkaleybar

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

This thesis presents a comprehensive dataset of trackers found on Dutch websites, particularly with domains hosted under the .nl Top Level Domain (TLD). The dataset has been further enriched by the categorisation af all websites, which enabled the in-depth analysis of how tracking activity and actors may differ within different categories of websites. This research aims to accurately illustrate the current web tracking landscape of the Netherlands.

The practice of tracking users on the web has seen a rapid growth over the past few years. While corporations amass large amounts of data to better understand and monetise user behaviour, through tools like targeted advertisements, concerns also grow about data protection and privacy in the global community. As stricter data protection rules have come into effect in the EU to address these concerns, to assess effectiveness and compliance, it is imperative to continuously monitor the tracking practices on the web. Like most technologies, web tracking is also becoming more advanced over time, able to circumvent modern privacy practices. A further complication is that different geographical regions might inhibit distinct tracking activities; therefore, the need for large-scale, country-specific datasets and studies arises.

The focus of this work is on compiling an extensive dataset to serve as a snapshot into the current state of the art in terms of web tracking in the Netherlands. The data consists of around 2.4M websites that were scraped for external links, and the found links were then cross-referenced with a known trackers database to identify potential trackers. Additionally, to further enrich the data, all websites in the dataset have been categorised using a pre-trained classification model. After the data collection, a comprehensive analysis was conducted to identify patterns and practices specific to Dutch websites. In line with the expectations, the analysis confirmed a significant tracking activity in the Netherlands and, furthermore, that there is a clear dominance of trackers owned by a few large, global corporations, such as Google and Meta. The dataset and the code are made available for future research to encourage further analysis in this field.

Contents

1 Introduction

Web tracking is the act of monitoring and recording user behaviour on the web. In recent years, this practice has become widespread across the web as it is a very effective tool for tracking engagement and predicting user behaviour. Among many other use cases, data collected from web tracking can be used to personalise web content and deliver targeted advertisements, making it highly effective for large-scale marketing campaigns. Web tracking can be used to benefit everybody by improving user experience and optimising advertisement revenue at the same time. However, tracking practices are often opaque and provide very little user control; therefore may raise concerns over user data exposure and even the possibility of influencing user behaviour.

Despite recent regulatory efforts to protect personal data, such as General Data Protection Regulation (GDPR), there has not been a noticeable improvement in terms of transparency and user agency. In fact, recent research that looked into the effect of the GDPR regulation has found the impact to be limited, with many websites being non-compliant or finding legal workarounds (Kretschmer et al., 2021 [18]). Often, websites track the user before consent is provided, and in some cases, tracking may even intensify after the user rejects consent (Papadogiannakis et al., 2021 [26]).

To effectively address the implications of web tracking, the current practices must be better understood. This can be achieved through analysing data and continuous monitoring activities and tendencies across the web. A major challenge is the unavailability of web tracking data; therefore, there is a need for up-to-date datasets with reproducible data collection methods. Although a general understanding may be established through analysing global data, it is important to conduct more geographically specific research, as the tracking landscape may differ in different countries or regions. While research is already available on a global scale, there are gaps in terms of large-scale, country-specific studies. To address these, scalable, systematic approaches must be developed to collect, categorise and analyse country-specific tracking data. This research, which focuses on tracking data within the .nl TLD, aims to fill this gap by developing a dedicated tool, compiling a large-scale dataset and providing an in-depth analysis of the Dutch web tracking landscape, answering the following research questions:

- **RQ1**: What is the prevalence of third-party web tracking on Dutch websites within the .nl TLD?
- **RQ2**: How does the presence and type of trackers vary across different categories of Dutch websites?
- **RQ3**: Who are the dominant tracking entities (local vs. foreign) operating within the Dutch webspace?

This thesis investigates tracking activity on websites with domain names registered in the Netherlands under the .nl TLD. While websites may be hosted elsewhere than the domain registration, filtering websites based on actual location is not a straightforward practice, and on the scale of this research is impractical. At the same time, it can be assumed that websites with a Dutch domain registration are intended for users in the Netherlands; therefore, it is sufficient to apply filtering based on the aforementioned criteria.

The data has been collected from live websites through static analysis of Hypertext Markup Language (HTML), Cascading Style Sheet (CSS) and JavaScript (JS) files, where trackers are identified by matching third-party links to known tracker domains. While this is not enough to create a complete picture of the whole web tracking space within the Netherlands, the gathered insights are nonetheless meaningful as the final dataset comprises almost a quarter of all Dutch domains registered and with an active website, making it representative of the whole population.

The contributions of this research include developing the Tracker Scraper utility for crawling websites and extracting third-party trackers; a large-scale dataset of categorised Dutch websites including identified trackers; and an in-depth analysis providing insights into the tracking landscape in the Netherlands. The data and the Tracker Scraper programme, including detailed documentation, are made available alongside this thesis.[1]

With the purpose of compiling a meaningful, large-scale dataset to enable in-depth analysis of the Dutch tracking space, this study aims to establish a baseline for regional, static collection and enrichment of web tracking data. This can be used in further research to truly understand overall tendencies within a certain location, to be able to effectively protect users against malicious data practices by spreading awareness and monitoring regulatory compliance.

While it may be expected that a country part of the European Union, and hence more regulated, would have a lower tracking activity, recent research (Kretschmer et al., 2021 [18]) suggests that there has been a major surge in web tracking that regulation might not be able to tackle fast enough. Furthermore, considering the recent technological developments, such as the large-scale adoption of cloud computing, the rise of Large Language Models (LLMs) and the increasing adoption of online tools based on these technologies, by users and companies alike, it has become even more imperative to understand how tracking occurs and what its implications are. While the cloud enables the collection and processing of virtually infinite amounts of data, generative Artificial Intelligence can further enhance the effectiveness of utilising this data and compiling it into meaningful statistics, creating a more comprehensive picture of the user's identity by linking data even between websites and platforms.

[1] https://github.com/daniellgelencser/tracker-scraper

The structure of this thesis is as follows: in the Background & Related Work (section 2), the notion of web tracking and its evolutions will be discussed, with a specific focus on understanding web tracking in the Netherlands. The next chapter will detail the Data Collection Methodology (section 3), which explains the collection of the initial set of URLs; the architecture, design and implementation of the Tracker Scraper utility; the website classification using the pre-trained Homepage2Vec model; the gathering of the known trackers list; and the structure of the database. As part of the Data Analysis (section 4), firstly, the model is evaluated in the context of the dataset, then a general overview of the gathered data is presented, and finally, a comprehensive Tracker Landscape Analysis is conducted to gain a clear picture of tracking tendencies in the Netherlands. This study concludes by discussing the implications of the analysis and the methods used in the Conclusion (section 5).

2 Background & Related Work

In the following section, the notion of web tracking will be discussed in detail, with a specific focus on how it works and its evolution, observed through related work. Furthermore, we address the impact of tracking on privacy and the progression of regulations to protect vulnerable users. The last section then explores the tracking landscape in the Netherlands and factors that may have influenced it over time.

2.1 Web Tracking

Third-party web tracking is the practice of monitoring and collecting data about users' online activities by entities that are not the primary website the user is visiting. This enables external parties, which include advertisers, social media or data analytics companies, to gather insights about users as they navigate the web.

To better understand the practice of tracking, it is important to know how interactions on the web generally happen. When browsing the web, there are three actors involved while interacting with websites: first is the user, often referred to as a client requesting a certain resource; second is the web server providing the website; and the third actor is any external entity providing additional content, typically called a third-party (Schelter and Kunegis, 2016 [27]). Communications between the client and the server are considered first-party interactions; however, when content is requested from a party external to the server, this becomes a third-party interaction. In a common scenario, a client would send an Hypertext Transfer Protocol (HTTP) request to the server where the website is hosted. The content of a web page arrives at the client's browser in the form of an HTML document, where the browser interprets and displays the different elements contained in the HTML document. This HTML document may instruct the browser to request further resources from the server, such as CSS, JS files or media content (images, videos, etc.). These contents can reside in the server, but can also be requested from external entities, where third-party tracking may be initiated.

Trackers may be embedded in websites in multiple ways, either statically loading at the same time as the website or dynamically loading after triggered events, such as clicking or hovering. Statistically embedded elements in the page can contain hard-coded links to third-party resources that can be used to track user behaviour. For example, these links to trackers might appear as a hyperlink or a resource, such as an image, that is loaded at the same time as the page. Furthermore, these elements may also be visually hidden from the user, such as invisible iframes or 1x1 pixel images. Dynamically loaded content, on the other hand, may be triggered by user interactions on the website, and the client-side script may call previously hidden links to trackers. Further methods can also include browser fingerprinting, where users can be uniquely identified based on information gathered on the client side, and third-party cookies, where information about the user can be stored on the client's side to enable cross-site tracking (Acar et al., 2014 [1]).

Generally, for a tracker to appear on a website, the owner or developer must embed it in the source code. One clear example of these is analytical trackers, where the website owner wants to gather analytics on how users access their websites. This is often implemented using a third party, such as Google Analytics. However, trackers may also be invoked in implicit ways, such as calling third-party resources distributed through Content Delivery Networks (CDN), hosting providers injecting third-party scripts or through Content Management System (CMS) themes and plugins, where owners may not even be aware of the tracker.

2.2 Review of Existing Literature

While the research on the topic of web-tracking is relatively new, the practice dates back to the 1990s when advertising and cookies were first introduced. Originally, the purpose of cookies was to enhance user experience on the website, such as remembering login information or shopping cart items, but it didn't take long before it became possible for third parties to be able to pay in order to place cookies on other websites and start tracking user behaviour. (Lerner et al., 2016 [21]; West, 2019 [35]).

The practice evolved and became increasingly widespread and commercialised in the years that followed, triggering concerns over user privacy and more attention from the research community. According to Lerner et al. (2016) [21], the first measurement studies into this field date back to 2005, although their longitudinal study finds evidence of web-tracking as early as 1996. A notable analysis of those early tracking activities is provided by Krishnamurthy and Willis (2009) [19], whose results show that between 2005 and 2008, the penetration of third-party tracking almost doubled in their dataset while the number of independent domains decreased. They also observe the trend of acquisitions by big companies and identify Google's penetration as nearing 60%.

Further studies (Hoofnagle and Good, 2012 [15]; Acar et al., 2014 [1]; Mayer and Mitchell, 2012 [24]; Acquisti et al., 2016 [2]) confirm the increase in tracking activity over time and expand the knowledge on the topic by analysing various aspects and implications. Online privacy is a particular concern that has been broadly examined by previous research. Hoofnagle and Good (2012) [15] discuss that cookies and third-party trackers are present in nearly all websites and how replacing old Flash cookies with HTML5 storage enables a more effective and less transparent tracking, thus raising concerns about user awareness. Research by Acar et al. (2014) [1] further showcases how tracking practices are becoming more sophisticated (e.g. canvas fingerprinting, cookie syncing) and provides evidence that these methods can even bypass privacy tools and can persist even when the user deletes the cookies. In an early study, Mayer and Mitchell (2012) [24] explored web tracking in light of the newly introduced Do Not Track (DNT) HTTP header and regulatory changes, where several larger corporations were penalised based on deceptive cookie practices or unauthorised data collection. Focusing on user awareness, Acquisti et al. (2016) [2] explain that users who are concerned for their privacy still fail to effectively protect their own privacy due to complexity and the lack of transparency.

While the early findings are insightful, a common limitation is their small datasets. However, with the increased availability of data through archives such as Common Crawl [6] and the Web Archive [16] and the increase in processing power of personal computers and internet speed, analysing large-scale datasets became feasible and affordable. Some notable large-scale studies include those by Libert (2015) [22], Englehardt and Narayanan (2016) [12], and Schelter and Kunegis (2016) [27], who use millions of domains and websites for their analysis. Overall, their findings confirm the invasive nature of web tracking as they find that the majority of websites include third-party trackers and that the majority of web trackers are dominantly owned by a few large companies (especially Google and Meta). They also shed light on the common practice of statically embedding links to third-party services into HTML, and highlight that third-party services like CDNs, advertising networks and analytical platforms often include trackers. Furthermore, they discuss the dark side of tracking when users are rarely informed about being tracked, and the privacy implications of collecting user data in such a manner, while highlighting the need for better practices and tighter regulation.

In a more recent domain-specific longitudinal analysis, Su et al. (2023) [31] showcase how tracking has intensified even in the education sector, where users are part of more vulnerable groups, especially minors. Their findings show that between 2012 and 2021, tracking on educational websites has even surpassed the sharply increasing number of trackers in other, non-educational websites, specifically in terms of commercial tracking (e.g. social media, advertising).

A consistent finding of the academic research is that web tracking has become more widespread, complex and opaque, increasingly capable of circumventing user consent and privacy measures (Cebere and Rossow, 2024 [4]). At the same time, tracking activity has increasingly been dominated by a small number of actors, further heightening the risks to user privacy. Despite warnings in early research concerning transparency and data protection, actions taken by regulatory bodies to limit user exposure stayed limited and reactive. While tracking technologies are becoming more sophisticated, the effectiveness of existing countermeasures (e.g. opt-out mechanisms, browser privacy controls) has diminished, highlighting the need for renewed effort to protect users.

2.3 Web Tracking in The Netherlands

In the context of the Dutch webspace, research on web tracking is more limited. An earlier study by Van der Velden (2014) [33] looks at tracking activities of Dutch government websites following the implementation of the EU e-Privacy Directive and finds the majority containing third-party elements, with big players such as Google and social media companies having a dominant share. Leenes and Kosta (2015) [20] detail the failure of this earlier regulation by looking at 100 Dutch websites and finding that 87% of them install cookies before the user is informed.

During the implementation of the GDPR, Dutch authorities were focusing on informing and advising, as opposed to issuing fines like other EU nations. Because of this policy-neutral approach, the number of fines was relatively low and the penalties were significant in the early stages of GDPR enforcement in the Netherlands. This trend has continued even two years after the regulation came into effect, with fines only being issued in very serious cases of misconduct. This shows a consistently cautious enforcement strategy, while other EU countries have issued a higher number of more substantial fines (Wolff and Atallah, 2021 [36]). A more recent large-scale study by Bouhoula et al. (2024) [3] further confirms that GDPR compliance is still, after seven years, inadequate, with a large proportion of websites still having privacy violations, such as visually biased consent buttons, ignored consents and missing cookie notices.

While users of the web are becoming increasingly more vulnerable to malpractice globally (Zac et al., 2023 [37]), as well as in the Netherlands, web tracking entities have mostly successfully evaded fines and prosecution, with non-compliance still being widespread. Although educating users is the first step, it has been shown that it is inadequate when dealing with modern, more advanced tracking techniques. Therefore, stricter regulation and enforcement are needed to protect users' right to privacy.

3 Data Collection Methodology

This section presents the methods used in compiling the web-tracking dataset. The dataset consists of a list of URLs and external links contained within the websites. The initial list of websites was acquired from the Common Crawl [6] archives from 2024 and classified using the Curlie [7] categories. The external links were retrieved using webscraping and then cross-referenced against a known tracker's database to identify trackers. The data collection was conducted from March to May 2025 and followed three major steps. Firstly, a comprehensive list of URLs of .nl websites has been gathered. In the second part, website content was collected and analysed to find the third-party links. Finally, the data has been enriched by categorising websites using a pre-trained model and categorising trackers based on the Ghostery dataset [14].

3.1 URL Collection

Compiling a list of websites presents a challenge on its own. Depending on the availability of the data, this could be easily solved, or it can become a tedious task of crawling websites ourselves. The most straightforward way of obtaining a website URLs is to use a list of registered domains from the authority owning the TLD. These lists are called TLD zone files and are sometimes made public; however, in most cases – including in the Netherlands – access to these files is restricted due to privacy concerns. For this reason, many researchers choose web-crawl data as a basis for these kinds of datasets. An advantage of web-crawl data compared to zone files is that web-crawl data consists of URLs of actual websites, while zone files include all registered domains, even if there is no website associated with them. Therefore, it is more suitable for web scraping purposes.

There are multiple ways of acquiring web-crawl data. These range from web-crawling using openly available libraries or writing the code ourselves to downloading the data from a trusted source. In the scope of this research, readily available Common Crawl [6] datasets were used to compile the initial list of URLs. Common Crawl [6] is a non-profit organisation that regularly collects crawl data and has one of the largest free datasets, geared specifically towards research. They provide various methods to access and query crawl data, including APIs for ease of access and file downloads to support large-scale datasets. For this research, the URLs have been extracted by downloading index files, as recommended by the official documentation for large amounts of data, such as a whole TLD.

The data has been retrieved from Common Crawl [6] through downloading crawl data from all the web crawls conducted in 2024 (Appendix A). The URLs of the crawl data are contained in multiple index files. The index files are obtained in GZIP compressed format, as each of these files spans several gigabytes. After unpacking the files, the content must be filtered due to the possibility of multiple TLDs being present in the file. Generally, the index data is ordered by TLD before it is split into separate index files; therefore, when filtering for a single TLD, the first and the last file will likely include entries with different TLDs.

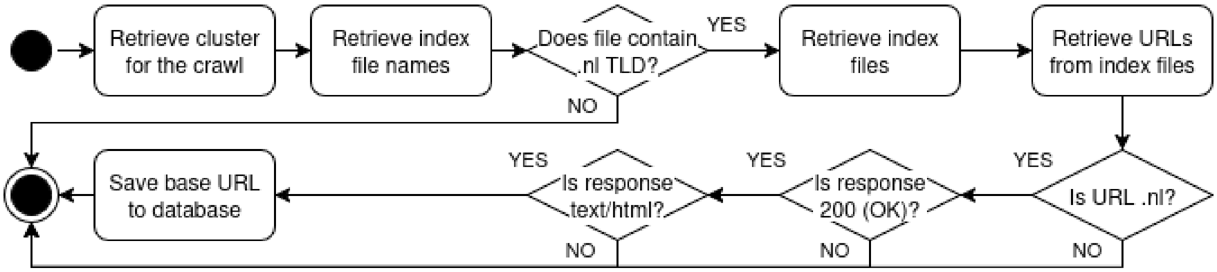
Retrieve cluster for the crawl
Retrieve index file names
Does file contain .nl TLD?
YES
NO
Retrieve index files
Retrieve URLs from index files
Is URL .nl?
YES
NO
Is response 200 (OK)?
YES
NO
Is response text/html?
YES
NO
Save base URL to database

Figure 1: Tracker Scraper: URL Collection Data Flow

As a part of this research, a Python script has been developed to gather the website URLs from the Common Crawl [6] archives using the aforementioned method (figure 1). The script retrieves the specific index file for the specific crawl, then iterates through all the index files containing the .nl TLD, filtering out other TLDs and gathering the relevant URLs. Index files contain a JavaScript Object Notation (JSON) string representation of the crawled resource containing the URL itself, the mime type of the resource and the status code of the response from the URL. To make sure only accessible and valid websites are collected, URLs have been filtered by OK response codes (200) and HTML data type (text/html).

URLs from separate index files are processed and saved in separate batches to minimise data loss when encountering an error. Uniqueness of the data is ensured by enforcing a unique index in the database. When attempting to insert a duplicate record, the script will keep the original record and disregard the incoming duplicate to prevent accidental data deletion. The script stops when all the index files have been processed within the single crawl. During the data collection, URLs from 10 different crawls have been collected to compile an initial dataset of more than 2.4 million URLs.

Furthermore, the script processes URLs after parsing and extraction from JSON format, by truncating paths and query sections of the URLs, so only the protocol (HTTP/Hypertext Transfer Protocol Secure (HTTPS)) and the domain name (including all subdomains) parts are left. As can be seen in the Web Scraping section (Section 3.2), we only collect the main page (if it exists) for each domain to avoid overwhelming the websites with multiple requests.

The URL collection script uses the following Python libraries: the *requests* library for making HTTP requests to retrieve the contents of the cluster and index files; *json* for parsing URL record contents from the index files; *re* to split lines of the cluster and index files using regular expressions; *sqlite3* for inserting URLs into the database and *gzip* for unpacking compressed index files.

3.2 Web Scraping

The second and largest part of this research is scraping the websites for each previously collected URL, looking for third-party links and possible trackers. To achieve this, an efficient script needed to be developed to collect trackers in a reasonable timeframe. The program makes HTTPS requests for each URL and then searches for external links embedded in the page or related resources. There is only one request for each domain (or subdomain), to avoid overwhelming websites; therefore, only pages loaded at the root path (usually index.html) of the domain are considered. Furthermore, the script only extracts links from static resources; dynamically loaded links and trackers are outside of the scope of this research. The text content of the pages is also saved for reference and further analysis.

3.2.1 Architecture

The structure of the program is organised into a main script and three modules handling different kinds of operations. The database manager module is responsible for reading from and writing to the database. For example, it may retrieve the next URL to be processed from the database and, once the processing has completed, it would write external links back to the database. The file manager module is responsible for saving raw website data to storage and maintaining log files. And finally, the web scraper module is responsible for making HTML requests and processing the content of the responses, including searching for external links. These three modules are utilised by a main script that is responsible for the module setup, as well as passing the data between modules and coordinating the flow of operations. This kind of architecture promotes modularity and reusability, where each component can be utilised on its own without modules being dependent on each other. Furthermore, it also adheres to and supports the single-responsibility principle.

The program code follows an asynchronous programming model as this is well-suited for Input/Output (I/O) intensive operation, such as database or file read and writes, as well as making web requests. As opposed to synchronous programming, where long-running I/O operations would block the execution, this method allows us to keep on processing while waiting on data, enabling improved resource utilisation. To avoid too many read and write requests to the database, a read and a write semaphore have been implemented to limit concurrent operations. The degree of concurrency can be set in the configuration, as databases can have different limitations.

In this case, an SQLite database has been used as this is a lightweight and easy-to-use solution for projects where the data volume is too large for simple Comma-Separated Values (CSV), JSON or other text file formats. To improve concurrency, Write-Ahead Logging (WAL) mode has been enabled; however, the concurrency capabilities of SQLite are still limited as it only supports concurrent reads and not writes. Therefore, an asynchronous programming model was chosen to interface with the database instead of using multiple threads. In the case of more robust databases such as MySQL, a multithreaded approach may be more time-efficient. The program developed for this thesis has the potential to be extended to support other databases.

3.2.2 Data Flow

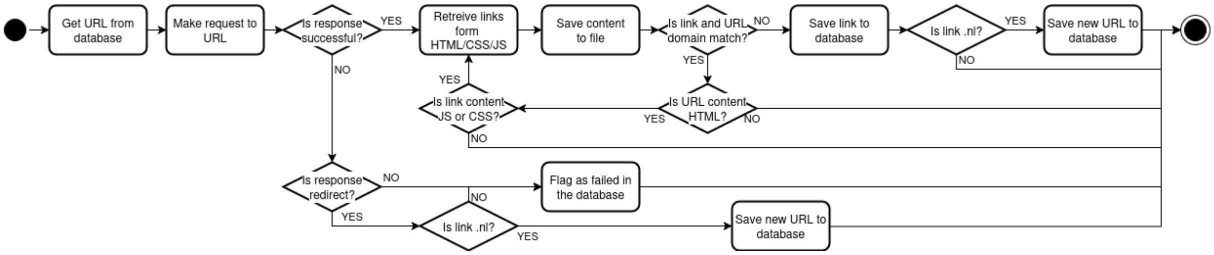

Figure 2: Tracker Scraper: Web Scraping Data Flow

The main program flow (figure 2) utilises multiple parallel tasks – still running sequentially – to retrieve and process data. Firstly, a single task is created for fetching URLs from the database and appending them to a queue. This task will also create a locking flag in the database to prevent processing the same URL multiple times. As the next step, a configurable number of tasks are created to pick up and process the URLs from the queue. These tasks will start with making an HTTP request to the destination of the URL and processing the response using a web scraper module.

The web scraper module provides separate functions to find links in HTML, CSS and JS files and then returns the results uniformly. If any links are found in the response, the response is then saved to a file. Then, the links are filtered based on whether they share the domain with the original URL or if they are external links. As we are interested primarily in external links, links to local JS and CSS files found in HTML files are saved to the database and processed separately. Further links to JS and CSS files, from style sheet and script resources, are not considered, as those are regarded outside the scope of this research. The third-party links are then saved to the database.

As an additional functionality, a mechanism has been developed to recycle third-party links in case they share the .nl TLD. With this, we can conduct web crawling to further extend the dataset until the desired number of results is reached or all the relevant, referenced URLs have been found.

3.2.3 Link Extraction

As the method of embedding links differs based on content type, distinct methodologies have been used to extract links from HTML, CSS and JS files. It is expected that the web scraper logic shall be able to detect most, if not all, embedded links. For the extraction from HTML files, the BeautifulSoup Python library has been used. This library allows easy navigation within the hierarchy of HTML tags. Links within HTML files are usually included as a value of a tag attribute. However, as scripting can be used to attach an actionable hyperlink to any attribute of any tag, we consider all attributes of all tags within a single page as potential URLs. The logic will attempt to parse attribute values as a URL. When parsing is successful, the link is identified; otherwise, the value is discarded as it does not contain a URL. Additionally, as discovered during testing, it is possible to include multiple links in a single HTML attribute; therefore, we attempt to parse each of the distinct links individually.

HTML embedded styles and scripts present in either attributes or tags are also considered, as well as separate JS and CSS files referenced from the main HTML page. To preserve storage and memory, both script and style files are first minimised, removing unnecessary content before they are processed and stored. Then URLs are identified using specific regular expressions to detect most possible embedded URLs within the text. This method is simpler than using parsing libraries while being comparably effective. Commented-out lines are also removed from the files, as regular expressions would consider URLs from comments as well. In all cases, we consider both absolute and relative links; however, relative links are only selected in case they point to a CSS or JavaScript file.

3.2.4 Error Handling

The script is developed to address errors efficiently. In case the user chooses to terminate the process using a Keyboard Interrupt, this will release locked rows in the database before the script exits. Furthermore, HTTP errors and redirects are propagated to the main script to run the appropriate database queries. In case of HTTP errors and redirects, the error code will be saved to the database, as well as flagging the record being failed, thus preventing it from being processed again. When a redirect happens, the target URL will also be added to the database, to be processed later on.

When a file operation fails, the record is released back into the pool of URLs to be processed; this way, it may be retried. In the event that there are no links found within the pages, but the request was otherwise successful, the record will be flagged as completed and successfully processed.

3.2.5 Limitations

Creating a picture of the full extent of online tracking is becoming increasingly difficult as tracking is getting more advanced, while detecting trackers is a complex and time-consuming task. Tracking identification is often done by searching references within a website to known trackers via matching the domain to external links. However, modern techniques, such as Canonical Name (CNAME) cloaking, can easily bypass this (Dao et al. 2021 [8]). To further complicate matters, certain parts of a web page may not be loaded simultaneously with the user's visit. This is called dynamic loading, a feature with the purpose of making websites more interactive. However, it can also be used to hide trackers by not delivering it as a part of the website's code, making it invisible for static analysis. To find dynamically loaded trackers, the webpage must be interacted with through a web browser or an advanced simulator, making large-scale data collection computationally very expensive, therefore infeasible within the scope of this research.

While the script can collect numerous trackers across different file types, our static analysis cannot capture dynamically loaded content. Although the script might not find all trackers, static analysis is still one of the best tools available for understanding large-scale tendencies in web tracking. As this method is less computationally demanding, it opens the possibility to gather large datasets which can be used to analyse trends in the tracking space. This data can be particularly useful for observing which actors engage in tracking activities and identifying the types of websites that are more likely to contain trackers.

Additionally, to avoid posing an extraneous load on websites, the script only makes a single request for each subdomain, with a limited number of retries; therefore, no data was collected from web pages other than the default webpage for that domain. With this in mind, login protection and cookie banners may have prevented data from being collected from certain websites. Coincidentally, some websites may also implement anti-bot measures, which can prevent the pages from being web-scraped by responding with error messages when suspicious activity is detected. This may have further reduced the number of successfully scraped websites in the dataset.

3.3 Website Classification

In the third and final stage of the data collection exercise, the dataset is enriched by assigning categories for each URL. As the known trackers' data is already categorised (Section 3.4), this provides a very useful final dataset where correlation between the website and tracker categories can be analysed in depth, providing valuable insights into current tendencies in tracking activity in the Netherlands. While manual categorisation of websites can provide high accuracy, due to the size of the dataset, with millions of websites, this approach is not feasible. Therefore, website classification, in this case, has been done using a Machine Learning (ML) model.

While most readily available ML models are trained on English websites, Lugeon et al. (2021) [23] present the pre-trained, language-agnostic Homepage2Vec that has been trained on 92 languages, including Dutch, to categorise websites. This model uses textual, visual, domain name and metadata-based features to render data into various categories. Furthermore, it is available as a Python library. The model makes use of the community-driven library, Curlie [7] (continuation of DMOZ), which is the largest human-edited, free directory of categorised websites. The Curlie [7] dataset includes 14 top-level categories, which are used by the model to classify websites.

To categorise the previously collected websites, HTML body and metadata, as well as the domain, have been used as features for the Homepage2Vec [23] model. While the model presents the option to use visual elements for classification, this feature was not used for two reasons: firstly, storing images, such as screenshots, for such a large scale dataset posed a challenge with constrained computing resources, secondly there is only slight improvement shown by Leugeon et al. (2021) [23] when using these extra elements. Results of the model are presented as predictions, including scores and embeddings. Scores include all 14 Curlie [7] categories with the corresponding probability that the website belongs to that category, while embeddings contain 100-dimensional vectors representing the webpage and can be used for further clustering and analysis. Within the bounds of this research, only probability scores have been used, as the model itself performed well enough to produce meaningful results.

3.4 Known Trackers

A main challenge in detecting tracking activity on websites is identifying which third-party URLs are actual trackers. While there is an abundance of trackers on the web, they are often hidden to avoid being found (Englehardt and Narayanan, 2016) [12]. Therefore, modern detection tools as well as recent research often use a pre-compiled and curated list (also known as a blocklist) of known trackers, such as EasyList [11], Disconnect.Me [10], WhoTracks.Me [14]. These databases are usually maintained by experts and communities and contain a list of known tracking domains, sometimes alongside categories and owner information (Snyder et al., 2020 [30]). Blocklists are often preferred to other methods of tracker detection as simple URL matching is significantly faster and has lower overhead compared to other methods such as heuristic algorithms, ML classifiers or behavioural analysis (Merzdovnik et al., 2017 [25]; Iqbal et al., 2022 [17]). While blocklists do not contain an exhaustive list of trackers, this is a robust, scalable and standardised way of identifying trackers (Fouad et al., 2018 [13]; Dimova et al., 2021 [9]).

To support tracker identification, this research utilises a combined blocklist sourced from

Disconnect.Me [10] and WhoTracks.Me [14], comprising 5686 trackers. The final blocklist is further enriched by categorising trackers and adding ownership information based on the data available from WhoTracks.Me [14]. However, as not all trackers in the combined blocklist are in the WhoTracks.Me [14] database, only a total of 4731 trackers have categories, and 4220 trackers have ownership information. Trackers are identified by cross-referencing external links with the blocklist via domain matching. This data is then used in the analysis of the tracking landscape (Section 4.2) to showcase the tendencies in the tracking landscape of the Netherlands.

3.5 Tracker Database

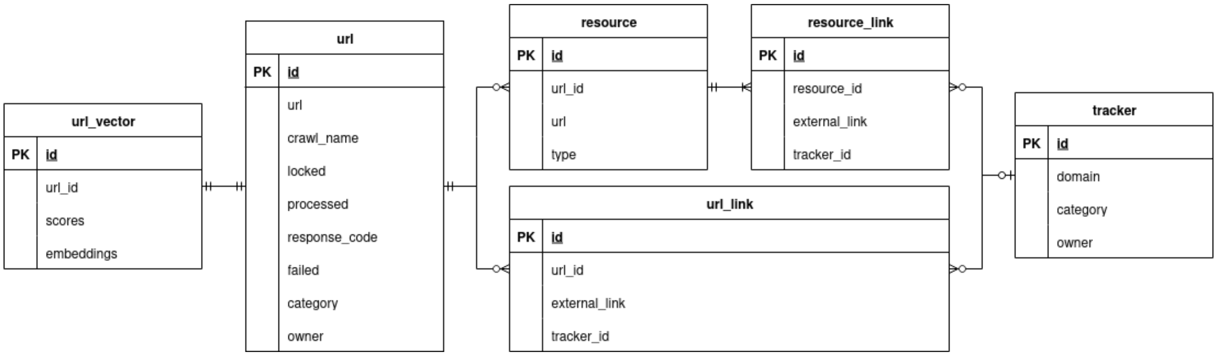

Figure 3: Entity Relationship Diagram (ERD) of the database

Figure 3 depicts the database schema used to collect .nl URLs, resource URLs to JS and CSS files, third-party links and the blocklist of trackers. The URL table contains all the collected URLs, including a reference to the Common Crawl [6] archive name, where the URL was collected from the crawl_name column. This column may also be empty if it was obtained by web crawling as part of this research. The locked and processed columns are used during the data collection, while records are flagged processed when a non-empty HTML page has been successfully received. The response_code column refers to the HTTP response number received from the website (200 when successful) alongside the failed flag indicating an error when retrieving the content from the website. Finally, the category column represents a single Curlie [7] category, which had the highest probability score as the result of the Homepage2Vec [23] model. This column is only used in the control set to evaluate the model. Local links to JavaScript and CSS resources are saved to the resource table, similarly to the URL table alongside the type identifying if it is a stylesheet or script.

The tracker table contains all tracking domains in the combined blocklist alongside categories and the name of the owner. The IDs of this table are used as a reference in tables url_link and resource_link for referencing trackers within the website. Both url_link and resource_link tables contain all the third-party links found within the website and follow a nearly identical table schema containing the external_link to the resource and the tracker_id if a tracker with the same domain has been found.

Results of the website classification (Section 3.3) have been saved in the database inside the url_vector table. The scores column contains the probabilities for each of the Curlie [7] categories the website received while evaluating with the Homepage2Vec [23] model. This data is used in the analysis part (Section 4) to observe category distribution in various statistics. The embeddings represent a 100-dimensional vector outputted by the same model and can be used in further improvement of categorisation. However, in this research, embeddings are not used specifically.

4 Data Analysis

This section presents a comprehensive data analysis that provides insights into the collected dataset. It starts with an overview of the whole dataset, describing some of the main characteristics of the Dutch website landscape and exploring the distribution of website categories within the Netherlands. This is followed by an in-depth analysis of tracker prevalence on different website categories, top global and Dutch trackers within the Netherlands, and the correlation between the different website categories and trackers found within them. At the end of the data analysis, the website categorisation model has been evaluated to confirm the correctness of the dataset and, therefore, validate the results of the data analysis.

4.1 Dataset Overview and Distribution

According to the Dutch registrar (SIDN [29]), there were 6.1M .nl second-degree domain names registered in the Netherlands, of which around 4.76M had an active website. The datasets presented in this research include 3.92M URLs of which 1.54M have distinct second-degree domain names. Around 2.46M websites have been successfully collected by retrieving valid HTML responses across 1.16M distinct domains. To our knowledge, the resulting dataset is the largest of Dutch websites to date and covers nearly a quarter (24.37%) of all domains with a live website in the Netherlands.

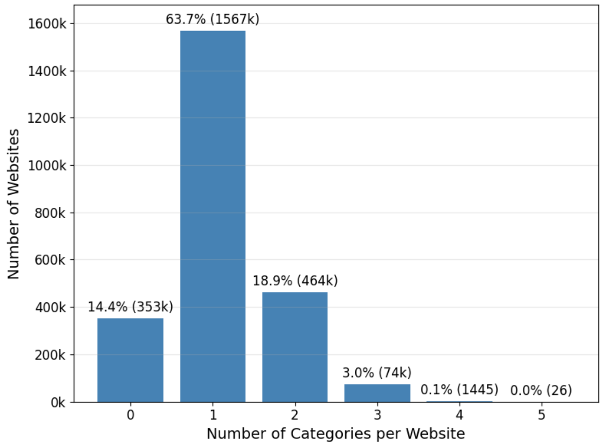

Figure 4: Distribution of Category Overlap

All of these websites have been categorised into the Curlie [7] categories using the pre-trained Homepage2Vec [23] model. A category is assigned to a website based on the strict criteria of having at least a 0.8 probability score in that category; therefore, some websites do not have a category, while others have more than one category. After discarding uncategorizable websites, around 2.1M websites remain for data analysis. This filtering further improves accuracy while still being representative of the whole .nl space.

Figure 4 shows the distribution of the number of categories across all collected websites. Websites with a single category dominate the dataset, just like in the Homepage2Vec research paper [23]. However, there are significantly more websites having two or more categories than in the original paper, where a decision threshold of 0.5 was used. The difference can be attributed to the fact that the dataset presented here is around 82 times larger, contains more noise, and is closer to the real-life composition of the .nl web. Out of all the websites having one or more categories, a website belongs to 1.29 categories on average in the presented dataset. In regard to websites with 0 category, the Homepage2Vec paper does not mention the exact number of uncategorisable websites, while they do mention that they only use data where at least one category is present. In line with this, in this research, further data analysis only includes websites with at least one category.

In the Category Distribution chart (figure 5), a significant difference can be observed between the Homepage2Vec research (Luegon et al., 2022) [23]. Compared to their global statistics, where the Business category takes the lead, our data indicates the Computers category to be the highest in the Netherlands. This result may be driven by the strong and growing IT and software sector in the region, as also noted by Ciff et al. (2024) [5] and by industry statistics (Weltevreden, 2024 [34]). It highlights how different countries and regions may have significant differences in the composition of the local websites. On the other hand, business is still in the top 3 categories, which is in line with Luegon et al. (2022) [23] findings; however, the share of the .nl dataset is significantly lower than their global average. The Home category being in second place is an interesting result, however, not entirely surprising considering the popularity of online shopping and the growing number of e-commerce websites within this category (Thuiswinkel.org, 2025 [32]).

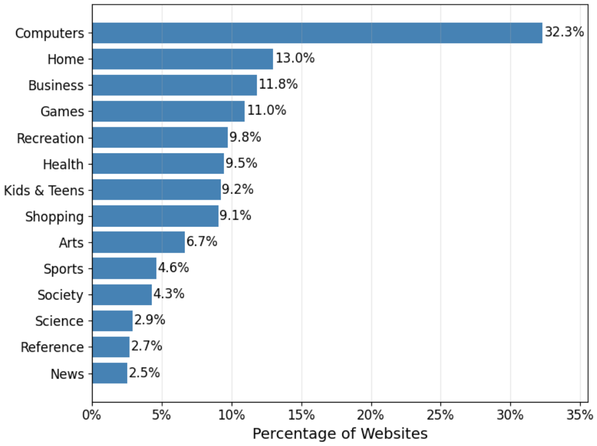

Figure 5: Website Category Distribution

Filtering any kind of content, regardless of whether it is illegitimate or unethical, can result in a skewed dataset that is not an accurate representation of the real-life web. The Curlie [7] dataset is a moderated dataset that excludes individual product promotions, marketing schemes and illegal content and separates adult websites in a restricted category. As the Homepage2Vec model was trained on this dataset, these specific contents are excluded, and therefore, the model cannot accurately categorise these websites. The data in this research is unfiltered and therefore includes such content types, which resulted in about 14.4% of websites not clearly belonging to any category. As it is unclear how much of the uncategorizable data belongs to the restricted content types, websites with ambiguous categorisation, where none of the categories scored higher than 0.8, are disregarded, thus may implicitly filter out some of the restricted content types.

Despite filtering out a significant proportion of the data, the resulting dataset is still large enough to represent the real-world Dutch web composition. By filtering out low-accuracy results, the improved accuracy can support a strong and sound basis for further analysis.

4.2 Tracker Landscape Analysis

This section will analyse static tracking activity and tendencies in the Netherlands. It will delve into the volume and the type of tracking for each website category, and it will also show a breakdown between different file types such as HTML, CSS and JS. Only statically coded trackers are included; dynamically constructed URLs and trackers loaded in dynamic ways are not covered by the scope.

4.2.1 Overall Tracking per Website Category

The Trackers to Links Ratio chart (figure 6, left side) shows the percentage of links within a website that are identified as trackers. Websites in the Arts category have the highest proportion of trackers, with just over half of the third-party links within the website identified as trackers. The Arts category is followed by the Health and Business categories, with only a few percentage point differences. Websites in the Computers category have the lowest tracking ratio of all the categories, followed by Shopping and Science. The chart shows relatively low overall variation from the average, confirming that significant tracking occurs regardless of the type of website.

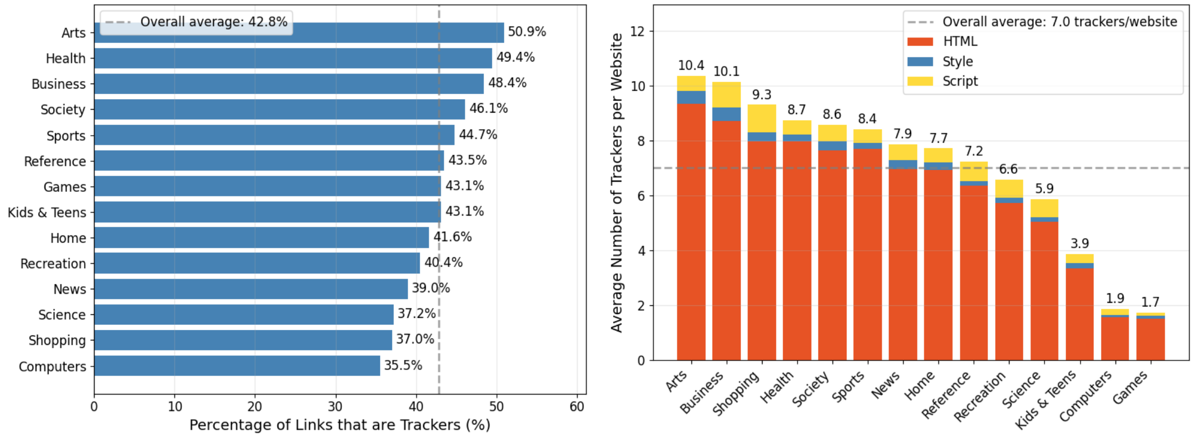

Figure 6: Tracker Penetration & Average Number of Trackers per Website

As it is shown in Figure 6 on the right, the highest number of trackers can be seen in Arts category websites, followed closely by Business and Shopping. The majority of the categories are close to the average of 7 trackers per website. However, we can see a significant drop in the number of trackers for the Computers and Games categories, where there are, on average, about 5 times fewer trackers compared to the highest categories. The stacked chart also gives insight into the distribution of trackers between different file types, which appears to be consistent between all categories. 88% of trackers found are contained within HTML, which shows that static tracking behaviour dominantly happens via links embedded in HTML. The lowest number of trackers is found in style sheets (3.6%), where embedding trackers is less straightforward. Trackers may appear in style sheets when third-party resources, such as images or other style sheets, are loaded, often via a CDN. Tracking in scripts seems to also be low (8.4%), although it mustn't be overlooked that there might be a lot more trackers loaded dynamically.

When compared with global statistics in previous research (Englehardt and Narayanan, 2016 [12]), several similarities can be observed, where Arts category websites have consistently high while Computers and Science category websites have consistently low tracking activity. However, there are also a few notable differences, for example, in the News category, a high number of trackers can be seen globally, while the Dutch data shows medium-to-low tracking activity in this category. Furthermore, the Business category appeared to have a low number of trackers globally, while having one of the highest numbers of trackers in the Netherlands.

There are several factors that may influence the high number of trackers on Arts websites specifically. Such websites can often include embeddings such as third-party media galleries, which may introduce trackers. Websites in this category are often created by individuals or small organisations with limited technical resources; therefore, they may choose to use CMS platforms and web builders. These platforms can often introduce additional trackers; for example, Wix and Automattic (WordPress) are two of the largest tracking entities. Furthermore, some websites may also offer free content; therefore, they may rely on third-party advertisements for monetisation, which is often associated with tracking activity.

In the case of the Computers category, the low tracking activity suggests that there is less tracking activity in the IT sector. These websites can often be built by highly skilled technical individuals with a better understanding of security principles and privacy awareness. Such websites often focus on a professional look with a minimalist design and the avoidance of third-party services. This may contribute to the more moderate tracking practices.

Interestingly, websites in the Games category have around average proportion of trackers to third-party links ratio while having the lowest overall tracker count. This shows that the utilisation of third-party links on these websites in general is relatively low. This may indicate that websites in this category tend to be simpler in structure, such as static websites, and therefore have less embedded content and fewer links present.

Overall, we can see some interesting facts about the web tracking landscape in the Netherlands; however, to better understand these results, further research is needed to identify the different tendencies within the different categories. Alternative categorisations should also be explored, as the structure offered by Curlie [7] might be too broad.

4.2.2 Top Trackers

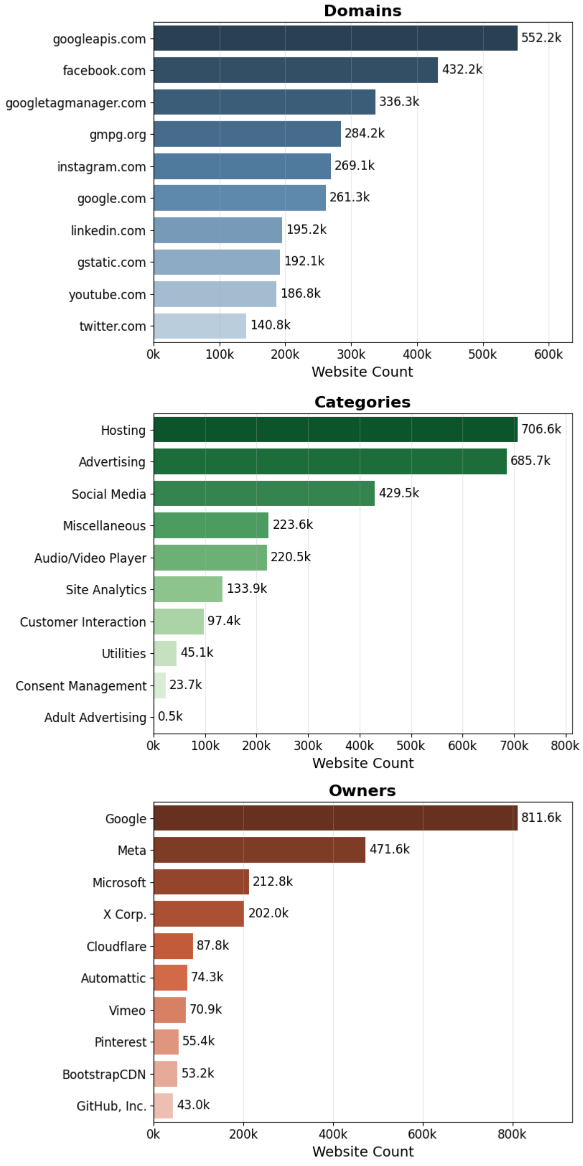

Figure 7: Top 10 Trackers. – *The horizontal axis denotes the number of distinct websites the specific tracker appears in.*

Further insights can be extracted by looking at the type of trackers and their ownership. Figure 7 shows the most common tracker domains, categories and the owning entities. Out of all the websites in the dataset, around 1.16M have been found to contain links to known trackers, which represents about half of the data. However, the actual number of websites being tracked is expected to be higher, considering that the data does not include dynamically loaded content, and there may be trackers that are not yet identified.

It may be observed that the top tracker domains appear in a significant number of websites, for example, googleapis.com has been found in almost half, facebook.com in more than a third and googletagmanager.com in more than a quarter of websites containing any trackers. This showcases that a few trackers dominate the tracker landscape in the Netherlands.

When analysing the categories of trackers found in websites, there is a clear dominance of hosting and advertising trackers, each contained in around 60% of websites with trackers, while social media trackers are found in over a third of websites. Although a large difference can be seen between categories, it is important to mention that adult websites may have been implicitly filtered out from this dataset, as detailed earlier; therefore, the number of trackers in the Adult Advertising category may be higher.

As expected, a large proportion of tracking activity is done by big American corporations such as Google, Meta, Microsoft and X Corp. Some of these large corporations own more than one of the most common trackers; for example, Google owns five of the top 10 tracker domains identified in the data, namely googleapis.com, googletagmanager.com, google.com, gstatic.com, and youtube.com. This is consistent with previous research, which found similar tendencies on a global scale (Englehardt and Narayanan, 2016 [12]; Schelter and Kunegis, 2016 [27]; Schelter, 2018 [28]; Lerner et al., 2016 [21]).

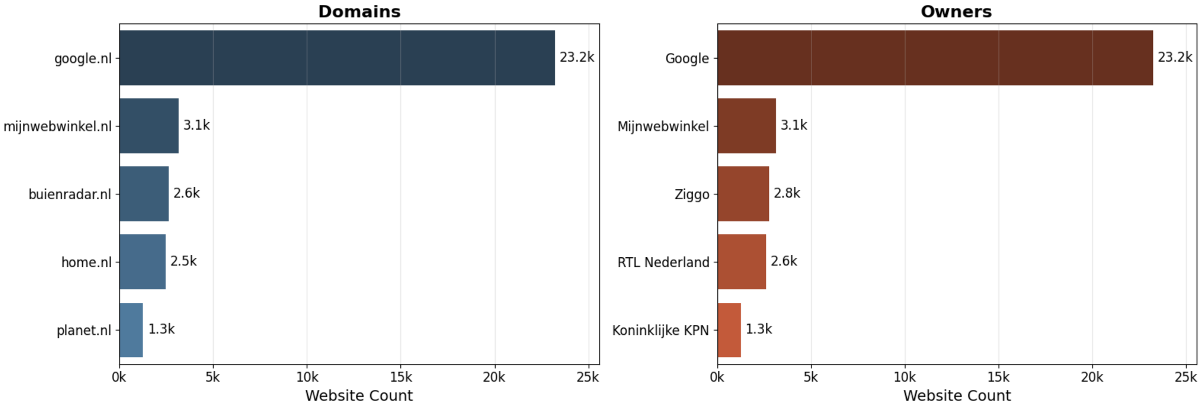

Figure 8: Top 5 Dutch Trackers. — *The horizontal axis denotes the number of distinct websites the specific tracker appears in.*

Looking specifically at Dutch trackers (figure 8), we observe that Google is still being overwhelmingly dominant with its local domain. The other top .nl trackers are represented by an e-commerce platform (Mijnwebwinkel), two major broadband providers (Zigo and KPN) and a large media company (RTL), although their occurrence is significantly lower. The most prevalent Dutch local entity (Mijnwebwinkel) still only appears in around 0.27% of websites containing trackers.

From these statistics, it is clear that while there are local trackers in the Netherlands, the web tracking space is still dominated by foreign entities. Out of all the major tracking entities, interestingly, only Google uses a local domain for tracking on a relevant scale. Using a local domain might instil trust with users that the data may be stored locally, and therefore it is compliant with local regulations; however, this is not necessarily the case as TLDs do not indicate data locality.

Overall, it can be observed that the web tracking landscape is overwhelmingly dominated by a small number of trackers owned by even fewer companies. This indicates that the immense amount of data collected via web tracking is highly centralised. Therefore, there is a geographical imbalance, as local tracking in the Netherlands is insignificant compared to the global market leaders.

4.2.3 Tracker Category Distribution

It can be seen in Figure 9 that trackers are not evenly distributed between website categories. Certain trackers appear in all website categories, while other trackers are concentrated in a single category. Looking at the domains chart, it is noticeable that certain trackers, such as cdn.shopify.com and cloudfront.net, are only in the top 5 trackers within the Shopping category, while google.com and jsdelivr.net are only in the top 5 trackers in the Computers and Games categories, respectively. On the other hand, googleapis.com and gstatic.com, both owned by Google, appear in the top 5 trackers for all categories. Websites in the Business category have the highest number of appearances of a single tracker, followed by Health and Home, with all three categories dominated by the gstatic.com tracker. The highest appearance of the googleapis.com tracker is also within the Business category.

When looking at the categories of trackers across categories of websites, a more even distribution can be seen, where 4 tracker categories, Advertising, Hosting, Site Analytics and Social Media, appear to be in the top 5 tracker categories for all website categories. On the other hand, the Utilities tracker category is only in the top 5 in one website category (Recreation). Trackers in the Hosting category have the highest occurrence across almost all categories except for Games websites, where Social Media trackers are dominant.

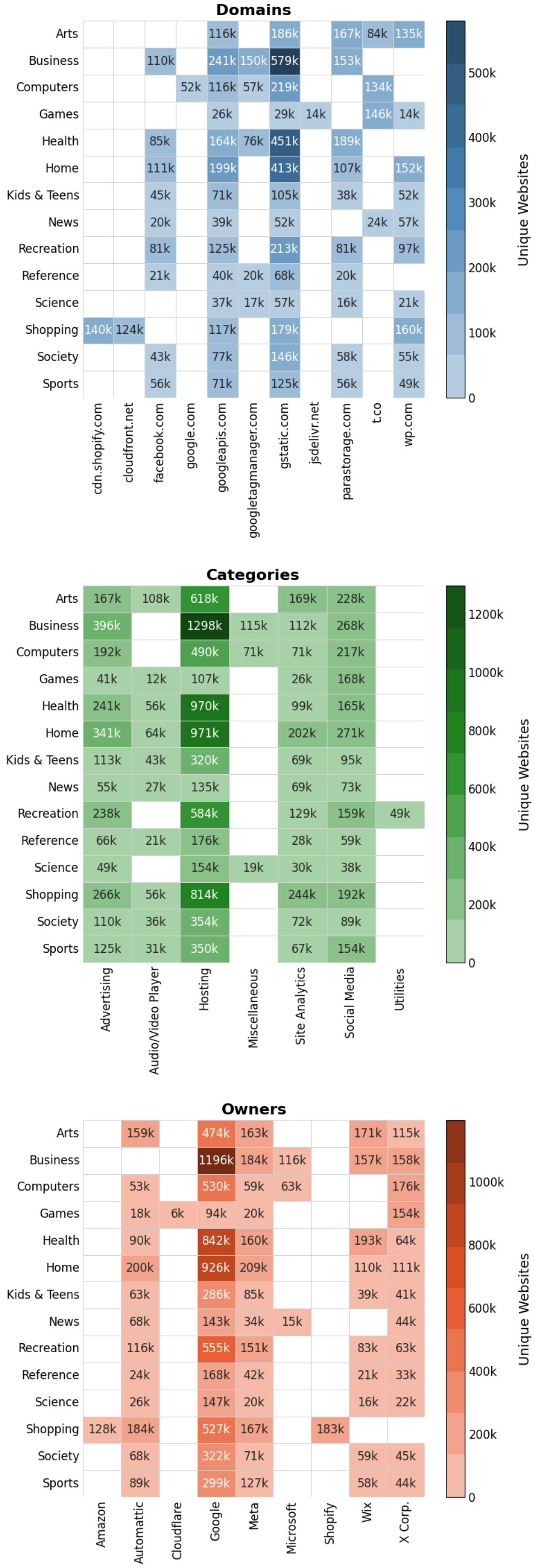

Figure 9: Top 5 Trackers per Website Category Heatmap

In terms of the tracking entities, the data shows that some entities focus on specific types of websites, while others are consistently within the top 5 across most or all categories. As expected, it can be seen that companies dominant in the e-commerce space, such as Amazon and Shopify, are within the top 5 entities within only the Shopping category. At the same time, Google and Meta are dominant in all website categories, while Automatic (WordPress) and X Corp. (formerly Twitter) are in the top 5 for all website categories except one. An interesting observation is that Microsoft only appeared in three categories among the top 5 trackers, even though it is the third-largest tracker entity in the Netherlands. Furthermore, a more privacy-focused company, Cloudflare, also appeared in the top 5 tracking entities for the Games websites.

Previously, in section 4.2.1, it was established that Arts websites have the highest tracking activity in total. However, when looking at the top tracker domains, categories and entities, a different tendency emerges where Business, followed by Home and Health category websites, have the largest appearance of a single tracker domain (gstatic.com), tracker category (Hosting), as well as tracking entity (Google). Analysing the distribution between website categories and specific trackers further confirms that Google and Meta are responsible for an overwhelming majority of tracking activity in the Netherlands.

4.3 Website Categorisation Model Evaluation

To support the data analysis presented, the validity of the collected data must be established. In particular, we evaluate the method in which the websites have been categorised using the pre-trained Homepage2Vec model (Lugeon et al. 2022 [23]). The model was designed for multilingual datasets, with Dutch websites included in their analysis. However, Lugeon et al. (2022) [23] used a dataset of only 29K websites to train their model. Therefore, there is a possibility that their training data was not representative of the dataset presented in this paper, which includes nearly 2.5M websites. Furthermore, evaluation is required as only HTML text and URLs were used to predict website categories, without the use of screenshots and pictures.

This was deemed to be a good enough representation, as the Homepage2Vec original paper [23] shows little improvement using visual content. The evaluation should also confirm the model's stability on potentially different content distributions.

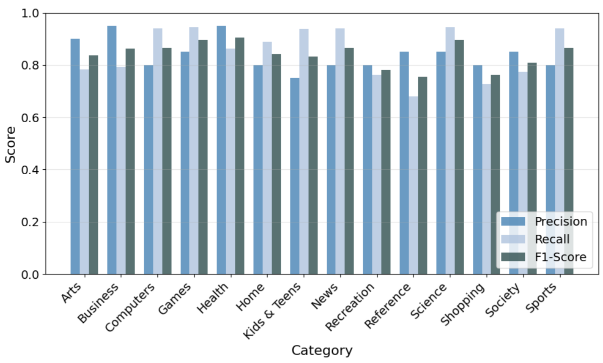

Figure 10: Homepage2Vec Model Performance Metrics by Category

To validate the Homepage2Vec [23] model's performance on the dataset presented by this research, a balanced approach was used. After all websites had been categorised, 20 websites were randomly chosen from each category, and the resulting validation dataset of 280 websites was labelled manually to establish a ground truth. For evaluating the performance precision, recall, F1-score, and Area Under the Curve (AUC)/Receiver-Operating Characteristic Curve (ROC) metrics have been calculated for each of the categories. From the results (figure 10), we can see that the model was able to predict the category of the websites correctly with a relatively high accuracy. It can be observed that the Reference category had the lowest F1 score at 0.756; however, this is still a reliable enough score. Out of the 14 categories, only 3 had a score below 0.8, and the Health category had the highest score with 0.905.

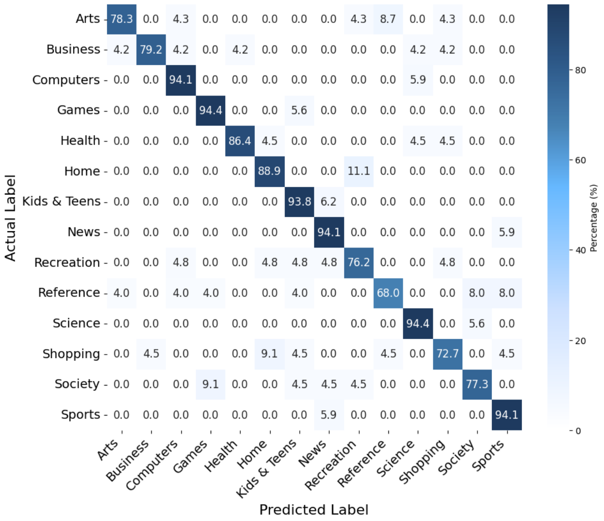

Figure 11: Website Category Confusion Matrix

The conduction matrix (figure 11) showcases that the false predictions were relatively low across all categories. Apart from a few outliers, the model was able to determine the right category for most websites. The Reference category is again the lowest contender at 68%, and Games and Science performed the best at 94.4%. The confusion is highest with category pairs that are harder to distinguish or overlap. The most prominent case was between Home and Recreation; however, these categories were harder to separate, even during the manual labelling. At the same time, the low overall confusion further confirms the strong performance of the model for the dataset.

The AUC scores show strong separability between the categories, with all scores about 0.83. The highest performers are again Games and Science at 0.966, which means these are the categories that the model can separate from other categories the best. The ROC curves shown in Figure 12 show that all the categories are fairly close to the perfect classifier and far from the random classifier, which validates the model's practicality.

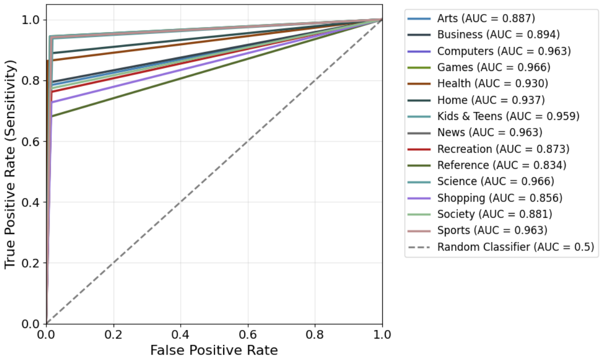

Figure 12: ROC Curves for Multi-Class Classification

Overall, we can see that the Reference category has scored the lowest in general accuracy and distinctiveness, which is likely since this is a broader category containing dictionaries, libraries, education and more; therefore, it can have a higher degree of overlap than other categories. While a high F1 score for Health shows that the results for this category are the most accurate, when it comes to distinctiveness, Games and Science take the lead. The overall F1 score for the dataset is around 0.84, which shows good accuracy, confirming that the model is suitable for categorising the dataset and that statistically sound assumptions can be made based on analysing the resulting data.

5 Discussion & Conclusion

In this thesis, as a result of thorough data collection, a comprehensive and representative set of .nl websites has been collected and scraped for all third-party links statically embedded in the site's source code. The data has been enriched by identifying known tracking entities based on the third-party links gathered and categorising all websites using a pre-trained model (Homepage2Vec). As it has been showcased in the Data Analysis (section 4), the final dataset enables an in-depth study of the tracking landscape in the Netherlands.

It may be further observed from the analysis that Dutch web tracking is concentrated on a few large foreign entities that are also widely identified as main trackers on the global scale by related research (Englehardt and Narayanan, 2016 [12]). At the same time, it can be seen that local tracking is by a magnitude smaller than foreign tracking in this country. Furthermore, the analysis highlighted that not all trackers are created equal. There is a clear distinction between specialised trackers, which are concentrated on a certain category of website, and generic trackers observed across the entire dataset. As an example, Amazon and Shopify trackers are only dominant in the Shopping category websites, whereas there are trackers by entities such as Google and Meta that seem to be dominant in almost all categories. This divide of focus is particularly interesting as there are similarly large corporations on both the generic and specialised sides.

While there is a consensus in research that an overwhelming majority of websites have trackers globally (Englehardt and Narayanan, 2016 [12]; Schelter and Kunegis, 2018 [28]), this study has found that only around half of Dutch websites within the dataset contain static trackers. This shows a substantial difference between .nl and other TLDs websites, suggesting that tracking activity, although still significant, is less prominent in countries with stronger data protection regulations. On the other hand, lower static tracking activity does not necessarily mean less tracking and could be a result of an evolution of tracking technologies where tracking companies focus on dynamic, server-side tracking and other more complex methods to avoid detection.

At the same time, tracking penetration is also lower than the global statistics reported in similar research (Englehardt and Narayanan, 2016 [12]), alluding again to the effect of tighter regulations. However, with 43% of third-party links identified as trackers on Dutch websites, web tracking is still significant. While the data seems to show improvements locally, there is still a need for reducing tracking, as popular websites are likely to be tracked. More specifically, the overwhelming majority of foreign trackers require extra attention, as they may raise questions on data locality and privacy.

While the collected data is comprehensive and a good representation of all .nl websites, it is not an exhaustive dataset; therefore, there may be missed outliers in the whole dataset, which might influence the results. Furthermore, only the home page has been scraped for every website — this was necessary to avoid overwhelming the servers with too many requests, meaning that further links and trackers may have been missed on other webpages. Sending one request per website also resulted in not getting past cookie banners, which (if best practices are followed) should have blocked most trackers, although previous research has shown that this is not necessarily the case (Papadogiannakis et al., 2021 [26]). Some websites may also employ anti-bot measures, which prevent web scraping; therefore, these websites are excluded from the dataset. Finally, as only around 50% of websites have been found to contain trackers, this may be due to the limitations of static analysis and could be mitigated by expanding to dynamic tracker detection. Nonetheless, at such a large scale, this would require significant computing power.

5.1 Future Work

In terms of future work, several aspects of this research can be extended or improved. While this work specifically looked at .nl websites, it does not cover all the websites in the Netherlands. To have a wider picture of tracking activity, websites within other TLDs can be added if they are hosted locally or intended for the Dutch audience. Furthermore, categorisation for both websites and trackers can be improved. A significant percentage (14.4%) of websites are uncategorisable using the Homepage2Vec model; however, this model also produces a vector representation of the website (also saved in the database), which can be used to improve categorisation using additional models. As different categorisation methods exist between different blocklists, not all trackers can be categorised using a single method. A further avenue for research could be proposing and developing a common categorisation method for trackers to improve statistics. Finally, follow-up studies could show how tracking activity changes over time and even between different countries. This monitoring is necessary as the fast-paced development of tracking technologies keeps reducing visibility and putting users' data at risk.

References

[1] Acar, G., Eubank, C., Englehardt, S., Juarez, M., Narayanan, A., Diaz, C.: The web never forgets: Persistent tracking mechanisms in the wild. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. pp. 674–689 (2014)

[2] Acquisti, A., Taylor, C., Wagman, L.: The economics of privacy. Journal of economic Literature **54**(2), 442–492 (2016)

[3] Bouhoula, A., Kubicek, K., Zac, A., Cotrini, C., Basin, D.: Automated {Large-Scale} analysis of cookie notice compliance. In: 33rd USENIX Security Symposium (USENIX Security 24). pp. 1723–1739 (2024)

[4] Cebere, B., Rossow, C.: Understanding web fingerprinting with a protocol-centric approach. In: Proceedings of the 27th International Symposium on Research in Attacks, Intrusions and Defenses. pp. 17–34 (2024)

[5] Ciff, T., Brouwer, A., Ponsioen, A., Van Lieshout, H.: Challenges and opportunities in the tight dutch it labour market. Technology in Society **77**, 102541 (2024)

[6] Common Crawl: Common crawl - overview. `https://commoncrawl.org/overview`, accessed: 2025-06-14

[7] Curlie: Curlie: The collector of urls. `https://curlie.org/`, accessed: 2025-06-14

[8] Dao, H., Mazel, J., Fukuda, K.: Cname cloaking-based tracking on the web: Characterization, detection, and protection. IEEE Transactions on Network and Service Management **18**(3), 3873–3888 (2021)

[9] Dimova, Y., Acar, G., Olejnik, L., Joosen, W., Van Goethem, T.: The cname of the game: Large-scale analysis of dns-based tracking evasion. arXiv preprint arXiv:2102.09301 (2021)

[10] Disconnect: Disconnect.me - freedom from tracking. `https://disconnect.me/`, accessed: 2025-06-14

[11] EasyList: Easyprivacy. `https://easylist.to/easylist/easyprivacy.txt`, accessed: 2025-08-23

[12] Englehardt, S., Narayanan, A.: Online tracking: A 1-million-site measurement and analysis. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. pp. 1388–1401 (2016)

[13] Fouad, I., Bielova, N., Legout, A., Sarafijanovic-Djukic, N.: Missed by filter lists: Detecting unknown third-party trackers with invisible pixels. arXiv preprint arXiv:1812.01514 (2018)

[14] Ghostery: Whotracks.me. `https://www.ghostery.com/whotracksme/`, accessed: 2025-06-14

[15] Hoofnagle, C.J., Good, N.: Web privacy census. Available at SSRN 2460547 (2012)

[16] Internet Archive: Internet Archive. `https://archive.org`, accessed: 2025-06-14

[17] Iqbal, U., Wolfe, C., Nguyen, C., Englehardt, S., Shafiq, Z.: Khaleesi: Breaker of advertising and tracking request chains. In: 31st USENIX Security Symposium (USENIX Security 22). pp. 2911–2928 (2022)

[18] Kretschmer, M., Pennekamp, J., Wehrle, K.: Cookie banners and privacy policies: Measuring the impact of the gdpr on the web. ACM Transactions on the Web (TWEB) **15**(4), 1–42 (2021)

[19] Krishnamurthy, B., Wills, C.: Privacy diffusion on the web: a longitudinal perspective. In: Proceedings of the 18th international conference on World wide web. pp. 541–550 (2009)

[20] Leenes, R., Kosta, E.: Taming the cookie monster with dutch law–a tale of regulatory failure. Computer Law & Security Review **31**(3), 317–335 (2015)

[21] Lerner, A., Simpson, A.K., Kohno, T., Roesner, F.: Internet jones and the raiders of the lost trackers: An archaeological study of web tracking from 1996 to 2016. In: 25th USENIX Security Symposium (USENIX Security 16) (2016)

[22] Libert, T.: Exposing the hidden web: An analysis of third-party http requests on 1 million websites. arXiv preprint arXiv:1511.00619 (2015)

[23] Lugeon, S., Piccardi, T., West, R.: Homepage2vec: Language-agnostic website embedding and classification. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 16, pp. 1285–1291 (2022)

[24] Mayer, J.R., Mitchell, J.C.: Third-party web tracking: Policy and technology. In: 2012 IEEE symposium on security and privacy. pp. 413–427. IEEE (2012)

[25] Merzdovnik, G., Huber, M., Buhov, D., Nikiforakis, N., Neuner, S., Schmiedecker, M., Weippl, E.: Block me if you can: A large-scale study of tracker-blocking tools. In: 2017 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 319–333. IEEE (2017)

[26] Papadogiannakis, E., Papadopoulos, P., Kourtellis, N., Markatos, E.P.: User tracking in the post-cookie era: How websites bypass gdpr consent to track users. In: Proceedings of the web conference 2021. pp. 2130–2141 (2021)

[27] Schelter, S., Kunegis, J.: Tracking the trackers: A large-scale analysis of embedded web trackers. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 10, pp. 679–682 (2016)

[28] Schelter, S., Kunegis, J.: On the ubiquity of web tracking: Insights from a billion-page web crawl. The Journal of Web Science **4** (2018)

[29] SIDN Labs: Sidn labs web statistics. `https://stats.sidnlabs.nl/en/web.html`, accessed: 2025-06-14

[30] Snyder, P., Vastel, A., Livshits, B.: Who filters the filters: Understanding the growth,

usefulness and efficiency of crowdsourced ad blocking. Proceedings of the ACM on Measurement and Analysis of Computing Systems **4**(2), 1–24 (2020)

[31] Su, Z., Helles, R., Al-Laith, A., Veilahti, A., Saxena, A., Simonsen, J.G.: Privacy lost in online education: Analysis of web tracking evolution. In: International Conference on Advanced Data Mining and Applications. pp. 440–455. Springer (2023)

[32] Thuiswinkel.org: Voor het eerst in jaren groei in online bestedingen aan producten (2025), `https://www.thuiswinkel.org/nieuws/voor-het-eerst-in-jaren-groei-in-online-bestedingen-aan-producten/`

[33] Van der Velden, L.: The third party diary–tracking the trackers on dutch governmental websites. NECSUS. European Journal of Media Studies **3**(1), 195–217 (2014)

[34] Weltevreden, J.: European e-commerce report 2024. Tech. rep., Amsterdam University of Applied Sciences & Ecommerce Europe (2024), `https://ecommerce-europe.eu/wp-content/uploads/2024/10/CMI2024_Complete_light_v1.pdf`

[35] West, S.M.: Data capitalism: Redefining the logics of surveillance and privacy. Business & society **58**(1), 20–41 (2019)

[36] Wolff, J., Atallah, N.: Early gdpr penalties: Analysis of implementation and fines through may 2020. Journal of Information Policy **11**, 63–103 (2021)

[37] Zac, A., Huang, Y.C., von Moltke, A., Decker, C., Ezrachi, A.: Dark patterns and consumer vulnerability. Behavioural Public Policy pp. 1–50 (2023)

Abreviations

AUC Area Under the Curve.

CDN Content Delivery Networks.

CMS Content Management System.

CNAME Canonical Name.

CSS Cascading Style Sheet.

CSV Comma-Separated Values.

DNT Do Not Track.

ERD Entity Relationship Diagram.

GDPR General Data Protection Regulation.

HTML Hypertext Markup Language.

HTTP Hypertext Transfer Protocol.

HTTPS Hypertext Transfer Protocol Secure.

I/O Input/Output.

JS JavaScript.

JSON JavaScript Object Notation.

LLM Large Language Model.

ML Machine Learning.

ROC Receiver-Operating Characteristic Curve.

TLD Top Level Domain.

URL Uniform Resource Locator.

WAL Write-Ahead Logging.

Appendix

A Crawl Data

Crawl Name	Number of (addditional) distinct URLs
CC-MAIN-2024-51	990,451
CC-MAIN-2024-46	148,246
CC-MAIN-2024-42	118,752
CC-MAIN-2024-38	106,456
CC-MAIN-2024-33	80,974
CC-MAIN-2024-30	98,921
CC-MAIN-2024-26	80,996
CC-MAIN-2024-22	64,403
CC-MAIN-2024-18	70,334
CC-MAIN-2024-10	95,351
Total	1,853,838