CO_2 and The Internet: a quantitative study on the environmental impact of the top Web sites in the world

Bachelor's Thesis

Vrije Universiteit Amsterdam

Supervisor: Ivano Malavolta

Second Reader: Kousar Aslam

Ivan Ivanov

2672214

Contents

Abbreviations

TWh	Terawatt-hour
MMT	Million Metric Tons
IEA	International Energy Agency
EPA	(US) Environmental Protection Agency
EIA	(US) Energy Information Administration
WRI	World Resources Institute
KWg	Kilowatt-hour per GB
TLD	Top-level Domain
CDN	Content delivery network
tCO2	Tons of CO2

1. Introduction

Awareness towards the threat of climate change has increased in the in the last years (Leiserowitz, 2019). Sustainability, a practice concerned with the recognition of the finite nature of resources, and their conscious usage with regards to future generations has also experienced a similar growth in recognition (EPA, 2021). As a result of this, there has been a large amount of CO_2-related scientific publications in the past years (Fabbrizzi et al., 2016).

CO_2 emissions have risen rapidly for more than a hundred years. Greenhouse gases, especially carbon dioxide (CO_2) emissions, are viewed as one of the main reasons of climate change, and it has slowly become one of the most important environmental problems for our world (Rehman et al., 2021). In the early 1900s there have been estimated to have been around 1000 MMT of carbon dioxide. In the 2010s these numbers reached almost 10,000 MMT (EPA, n.d.). This has been driven by a variety of factors: increased globalization, fossil fuel usage, population increase and more (Lindsey, 2020). According to the World Meteorological Organization's State of the Global Climate report, the global average temperature in 2020 was about 1.2°C above preindustrial level (World Meteorological Organization, 2021). Numerous attempts have been made to decrease the emissions. To mitigate the threat, the Paris Agreement called to limit global warming to below 2°C but preferably to 1.5°C, a number comparable to the previously measured pre-industrial levels. Additionally, there have also been public awareness campaigns and government reforms (Department for Business, Energy & Industrial Strategy, 2021) (Ge & Ross, 2019). The IT sector alone amounts to 1.4% of those global emissions but it can be reduced to 20% less of that if a switch to renewables were to happen (Telefonaktiebolaget LM Ericsson, 2020).

Along with the surge in climate change related concerns, there has also been a growth in global Internet data traffic. In the year 2020 alone web traffic exchange has expanded with an average of 35% in size (Krisetya et al., n.d.) and has been previously predicted to experience further expansion with close to five and a half billion internet users expected by 2023 (Cisco Systems, Inc., n.d.). This explosion in traffic growth is driven partially by the SARS-CoV-2 pandemic's effect, but also by the consistently upward developments in usage of Internet-connected devices from 2011 onward (Telefonaktiebolaget LM Ericsson, 2021, p. 3), and it is mostly caused by video streaming, conferencing, online gaming, and social networking (IEA, 2021). Device adoption rates vary widely per region, though it is estimated that in 2016 a United States citizen owned about 7.8 Internet connection-capable devices on average and consumes 97 gigabytes of data per month, several times more than the world average. Meanwhile a Chinese citizen had 2.5 and uses only 12 gigabytes. In the next two years, the statistics for the USA have increased to 10 devices and 140 gigabytes respectively (The Shift Project, 2019, p. 61).

As these devices become more and more of a central point in human lives, the Internet usage grows rapidly as well. 92% of internet users nowadays access the web with mobile phones (Statista & Johnson, 2022). Because of this, the amount of site hits grows larger with each year and that, in turn, affects the overall data consumption as well (United Nations, 2019, p. 15).

Data centers are the backbone of the internet. They consume around 1-1.5% of global electricity use (Masanet et al., 2020) which translates to an estimated use of around 200-250 TWh of electricity per year (IEA, 2021) with some estimates going as high as 400 TWh. In general, the electricity usage has fallen "by a factor of four" since 2010 due to hardware improvements in processor efficiency and idle power usage (Masanet et al., 2020). A report by the International Energy Agency (IEA) from 2014 stated that the development of energy efficiency metrics was one of three key considerations needed for effective policy

making and a reduction of the networks' energy usage and in turn, the carbon dioxide footprint (IEA, 2014b).

There has been research made about the IT sector's emission generation but despite the importance of ICT and the ubiquity of the Internet, but none of them have attempted to analyze the web's CO_2 footprint. In this thesis, an attempt will be made to create a clear overview of some of the web's most popular websites and the amount of carbon dioxide they are potentially generating.

To do so, the following research question will be addressed:

What is the current state of energy consumption of the top Web sites in the world?

Thesis Structure

In the next chapter a series of related papers are briefly examined. Chapter 3 talks about the research methods used and Chapter 4 focuses on the methodology and the formulas used to calculate the final numbers. Chapter 5 is an analysis on the received data and Chapter 6 discusses the results. In the last two chapters a reflection on the process of writing the thesis will be made and the last chapter is the conclusion.

2. Related Studies

Previously, separate studies have analyzed the electricity consumption of the average Internet data transfer for various devices (Thiagarajan et al., 2012; Zhu & Reddi, 2013). Others have examined the ICT sector's overall carbon footprint (Freitag et al., 2021). Direct estimate comparisons between any papers on this topic are difficult to do because of inconsistent usage of methodologies and uncertainty in the current accuracy of the data. Some are based on either estimates of regional or of worldwide energy consumption and footprint while on the web and are combined with traffic estimates to compute the amount of energy consumed and dioxide generated per some data amount. The distinction with the largest influence on the result is how the analysis boundaries have been set. Though some studies include the terminal equipment (e.g., personal computers and servers) within the system boundaries (Aslan et al., 2017), others do not. Because of these differences and the lack of access to more modern and detailed data regarding the exact electricity usage habits of modern data centers, a comparison of such statistics will not be undertaken here. Instead, this thesis focuses on data, which is particularly well characterized, and the system boundaries are clear and consistent. The goal of this paper is to examine in detail recently-collected data which reflect the state of the Internet and explicitly focuses only on websites.

3. Research Method

The purpose of this chapter is to explain the research methods used in this paper.

Methodology

The data analysis will be strictly quantitative. The goal is to analyze the impact of the internet surfing habits on a relatively large scale and to compare an innocuous habit with the real-world impact it has. Due to the nature of the service that the data is sourced from, the analysis is done entirely on the homepage of a given website and because of that there is no focus on analysis of common types of data transfer like streaming video from a particular streaming service, loading multiple pages from the same website or infinitely scrolling pages (e.g., Twitter, Facebook, NBCNews.com).

The main dataset used for analysis has been collected by using the Website Carbon API[1]. It is an online tool created by Wholegrain Digital that provides an estimate for the ecological footprint of a website. It awaits a query in the form of a URL address and returns a JSON file containing several statistics about the domain.

The dataset where all the statistics have been collected is a .csv file generated from the information calculated by the aforementioned API. It contains information about the 50,034 most popular websites and it has the following eight features:

1. Website URL
2. Type of hosting – Depends on the energy source used by the data center. Saved as either 'True' for websites hosted by a service provider using green energy or 'unknown' for those whose green status could not be determined. The status is determined by the Green Web Foundation's own API[2]. There, any websites that is hosted by a 'Green' data center is shown as using Green energy (Note: Not all centers mentioned on the GWF website use Green energy, in some cases they use standard grid and the emissions are offset afterwards. Additionally, the hosting status can sometimes not be detected dependent on whether or not the website is using a CDN. In such a case the IP address of the host cannot be identified and the CDN is deemed to be the host.). If the website is found to be using green energy, then the carbon emissions are reduced.
3. Number of bytes – The amount transferred upon the initial page load, provided that the website has not been visited before.
4. Number of bytes (adjusted) – An adjusted value for the second visit of a website which takes into consideration browser caching.
5. Percentage of sites it is cleaner than – A simple comparison between the amount of CO_2 the currently tested website generates and the others in the database.
6. Energy – The amount consumed upon a single page load, in Kilowatts per Gigabyte. This section takes into account the energy consumed by consumer device, data center, networks used at page load time and also during the time the hardware itself was produced. All numbers are estimates and are described in greater detail in "API Calculations".
7. Grams and liters of CO_2 generated by a renewable grid
8. Grams and liters of CO_2, by a standard national grid.

The website list used has been sourced from the Tranco[3] list of 1 million most popular websites. This is a list which uses averaged data from four other ranking providers (Alexa, Cisco Umbrella, Majestic and Quantcast). The reason for using Tranco, and not either of the four other rankings is that they have been found to often disagree on which sites are actually the most popular. Those lists can change daily and are manipulatable by third parties. The data has been sourced from the original rankings and then averaged over a thirty-day period (le Pochat et al., 2019). The list used to write this project is retrieved on 03/05/2022.

The dataset that is used for analysis has been sourced between the period of 04/05/2022 and 21/5/2022.

[1] https://www.websitecarbon.com/
[2] https://www.thegreenwebfoundation.org/
[3] https://tranco-list.eu/

Errors and Limitations

The set has been cleaned up of any accidentally repeated data and several manual edits have been made for the following issues:

1. Google redirects: Some URLs, like several of Google's regional domains and multiple other unrelated websites) redirect to either the main Google.com domain or one of the regional variations. This created several hundred duplicate "https://www.google.[region]/" entries. In this case the redirected entries have been removed and only the first one from each region has been kept with the presumption that it is the original one.
2. General regional redirects: The same issue appeared for several other websites and has been dealt with in the same way.

Additionally, the data has been formatted to account for visual clarity and ease of use (decimal sign placement, general formatting).

An occurring issue encountered during this data collection process is the fact that a large number of websites could not be analyzed (either correctly or at all) for several different reasons. These reasons are as follows:

1. The website was offline – Here the website returns a generic "site cannot be reached" message in the browser. This response is still being recorded in the dataset for reasons mentioned in the "Outliers" section of chapter 5.
2. The website was hosted by Cloudflare – In this case the API returned a response in an HTML form and not JSON. Each time that happens the corresponding website is skipped and removed from the dataset.

A few other criteria for being included in our dataset, as mentioned by Wholegrain Digital on their website, are:

1. The website is accessible through a standard web browser.
2. Login is not required.
3. Search engines are allowed.
4. Has content that is unique and aimed at human visitors. Meaning that, holding, error, server notification, demo and pages that are generally useless are excluded from this. (The last one is highly subjective).
5. Is free from any illegal/explicit materials

The original goal of the project was to parse the entire one million website list. Unfortunately, another limitation was encountered at the data collection process: the API used has a daily limit of 25,000 hits available and it is also shared with other users. Because of that the number of sites that could be parsed per day was no more than two to three thousand.

Overall, the first 65,600 websites from the Tranco list are parsed. 52,431 of those are actually processed (due to the issues mentioned above) and after the removal of any duplicates there are **50,034** usable websites left. The loss from parsed to parsable is 21.1% and from parsable to usable is an additional 4.6%.

Software

The research question and the scope of the paper were defined in the previous sections. Here, the tools used for the data collection and manipulation processes are described.

Essential software used:

- Visual Studio Code 1.68.1 (for Windows)
- Jupyter v2022.5.1001601848
- Python 3.10.4 64bit

There were two main tools used for data collection:

- An HTTP Request/Response program (get_requester.py) which main function is to send out asynchronous requests to the Website Carbon API and receives JSON formatted responses in return. It has been written with Python 3.
  Additionally, these libraries were used:
  I. ASYNCIO[4]: Part of the Python Standard Library. Utilizes concurrent programming concepts to make asynchronous programming possible. It is used because asynchronous requests are usually multiple times faster than sequentially programmed ones, depending on the implementation and the receiving server's limitations. Concurrency allows for different parts (e.g., functions) of a program to be executed independently. It means that executing functions can be done "in parallel" which significantly boosts the overall speed of execution of the program. Note that ASYNCIO does not truly support parallelism, but the execution of functions is so quick so that to the purposes of this experiment and the bandwidth of the data processed it works essentially the same.
  II. AIOHTTP[5]: Written by Nikolay Kim and Andrew Svetlov. A client/server library that utilizes ASYNCIO for making asynchronous HTTP GET requests targeted towards a public endpoint from the Website Carbon API.
  III. Throttler 1.2.1[6]: Used for throttling the amount of outgoing GET requests, as to not overload the receiving server.
- The Website Carbon API itself, which provides the data to the Python program.

And the following tool was used for the data manipulation and analysis process:

- An .ipynb notebook file (notebook.ipynb) which was used in VSCode to combine the collected .csv files into one file (called main.csv), format it, clean any potential issues with it and generate data for the thesis.
  Additionally, these libraries were used:

[4] https://docs.python.org/3/library/asyncio.html
[5] https://docs.aiohttp.org/en/stable/
[6] https://pypi.org/project/throttler/

I. Pandas 1.4.2[7] A Python written software library used for data science purposes. This is the library used the most in this project for reading files, plotting graphs, storing data and more.
II. NumPy 1.22.3[8]: Similar to Pandas, but mostly focused on array and math function handling. Used for a few functions in the notebook file.
III. Matplotlib 3.5.2[9]: A plotting library, written for Python as well and used for a few of the plots present in the thesis.
IV. Tld 0.12.6[10]: A small package which main function is to extract the top-level domain of the URL's present in the main dataset.

Python's 'pathlib', 'warnings', 're' and 'os' were used as well for miscellaneous purposes.

4. API Calculations and scope

Factors

The amount of energy and emissions generated by a webpage are calculated with the following factors[11]:

- Data transfer over the wire – the amount of data that is transferred from the server to the user upon a page load.
- Energy intensity of web data – The amount of energy used by data centers, networks, and the user's device to load a page. An estimated average is used here due to the many different variations possible when all devices and computers are considered.
- Energy source used by the data center – as described in "Type of hosting" previously.
- Carbon intensity of electricity – Carbon intensity is the amount of carbon dioxide generated to create a unit of electricity (nationalgridESO, n.d.). Usually measured in grams of CO_2 per kilowatt-hour. Here it is based on the international average for grid electricity.
- Website traffic – The number of page views on a website. Multiplying the amount of carbon generated per page view by the number of expected views for a website gives an overview of the actual impact a website has.

System Boundaries

System boundaries are defined by Klaus Büchel (1996) as what "defines the processes to be analyzed with regard to material and energy flows and emissions.". As such they act as "scope limiters" on the material that is being analyzed and contextualize the goals of the API.

To bring an accurate estimate to the energy usage of the network, the system boundaries have to be defined first. Defining them to be smaller in scope leads to a misrepresentation of the energy output and usage of the hardware involved here (data centers, networks, and end devices). If the opposite happens, the broadening would overestimate the amount of elements/hardware with any influence that need to be looked at and add unnecessary complexity.

[7] https://pandas.pydata.org/docs/reference/
[8] https://numpy.org/doc/stable/reference/index.html
[9] https://matplotlib.org/stable/api/index.html
[10] https://tld.readthedocs.io/en/latest/
[11] https://www.websitecarbon.com/how-does-it-work/

The boundaries set here for the different system segments are based on Anders Andrae's (2020) 'New perspectives on internet electricity use in 2030' study.

- Consumer device use: 52% End-users, 25% of those are returning visitors.
- Network use: The data that is transferred through the network. 14% of the system.
- Data center use: The energy used by the centers for operation. 15% of the system.
- Hardware production: An estimate for the energy used to create all of the devices taking part in the data transfer process. 19% of the system.

Additional estimates for the average energy usage of all devices have been derived for the purposes of this API, both from the Andrae study and from other sources as well. As far as carbon intensity goes, there is an average of 442g/kWh used here, which is sourced from Ember's Data Explorer[12].

The key metric used in the calculations is kWh/GB, or "kilowatt-hour per gigabyte" – the kilowatts per hour for each gigabyte of data transferred.

With all of that said, the Website Carbon API uses a number of formulas to make the exact calculations. They work as follows[13]:

Energy per visit in kWh (E):
E = [Data Transfer per Visit (new visitors) in GB x 0.81 kWh/GB x 0.75] + [Data Transfer per Visit (returning visitors) in GB x 0.81 kWh/GB x 0.25 x 0.02]

Emissions per visit in grams CO_2e (C):
C = E x 442 g/kWh (or alternative/region-specific carbon factor)

Annual energy in kWh (AE):
AE = E x Monthly Visitors x 12

Annual emissions in grams CO_2e (AC):
AC = C x Monthly Visitors x 12

Annual Segment Energy:
Consumer device energy = AE x 0.52

Network energy = AE x 0.14

Data center energy = AE x 0.15

Production energy = AE x 0.19

Annual Segment Emissions:
Consumer device emissions = AC x 0.52

Network emissions = AC x 0.14

Data center emission = AC x 0.15

[12] https://ember-climate.org/data/data-explorer/
[13] https://sustainablewebdesign.org/calculating-digital-emissions/

Production emission = AC x 0.19

5. Data Analysis & Results

Overview

Dataset Distribution

To provide a sound picture of websites and their CO_2 emission impact, more than 65 thousand websites are explored. All are sourced from the original Tranco list and there are no other distinctions done to the website's importance during the data collection process other than its ranking in the list.

Top-Level Domains: There are 50,034 usable websites in the final dataset. From those, 27,873 have the ".com" top-level domain, 3868 have ".org ", and 2065 have ".net " as the domain. Part of the distribution follows in Figure 1. Overall, there are 716 different top-level domains (Appendix A). Keeping only those which occur more than 100 times leaves us with exactly 40 domains. The distribution is heavily skewed towards the first 3 TLD's which account for 67.5% of the entire dataset.

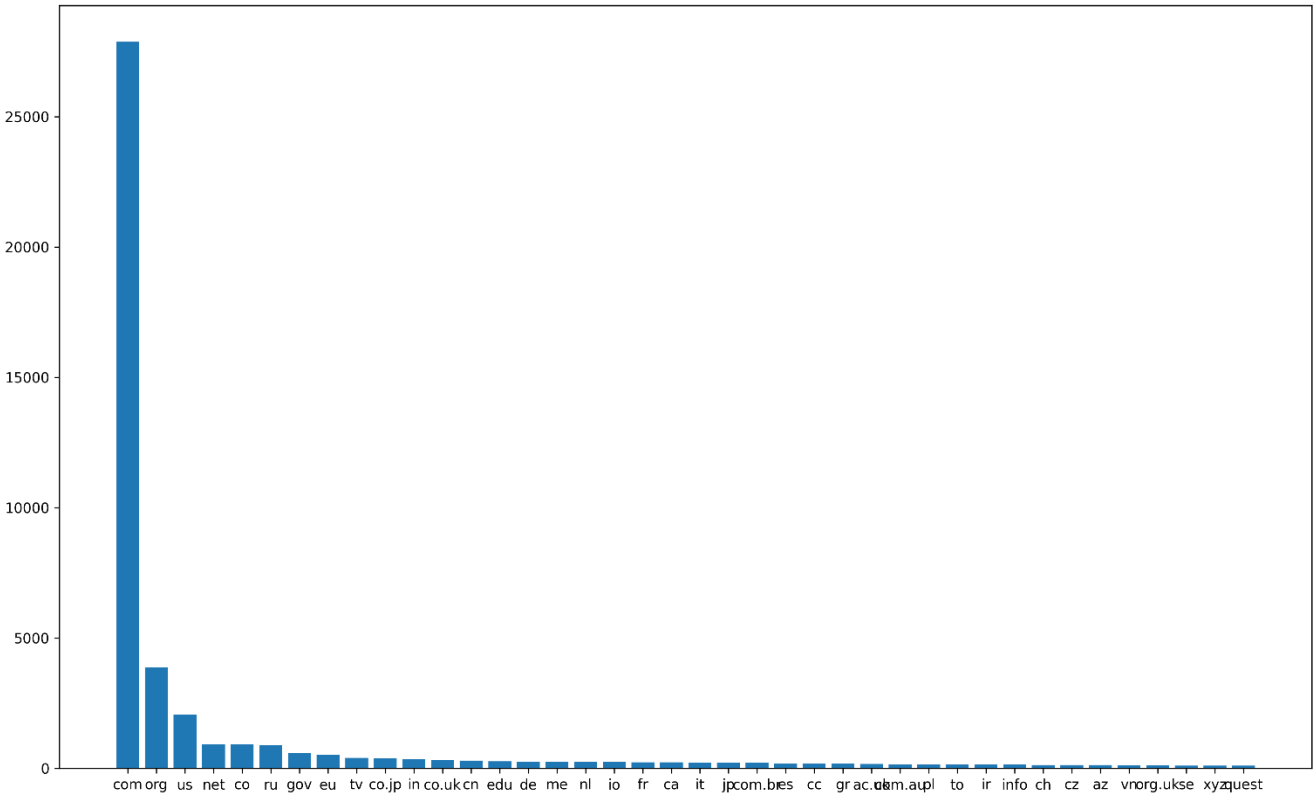


Figure 1: Partial TLD Distribution, first 40 TLD's

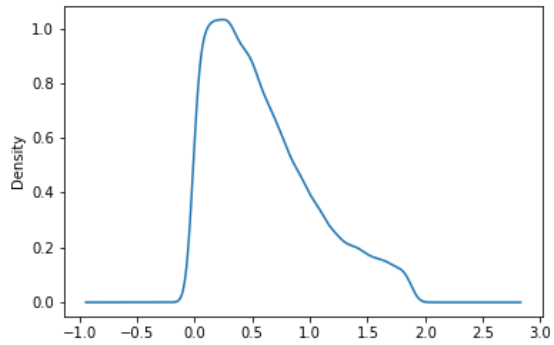
Website Sizes: A website's size might affect the time it will take to load it and the amount of CO_2 it will generate. The collected websites vary greatly in range. Looking at the size on an initial load, the smallest measured domain response weights at exactly 168 bytes[14] and the largest one is lematin.ma[15] at 304MB.

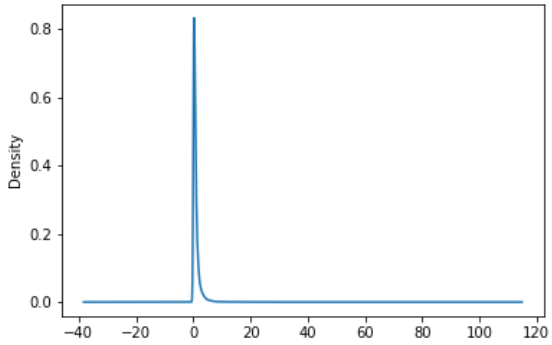
[14] http://bloog.pl/
[15] https://www.lematin.ma/

These are only outliers though as the mean size is only 3.69MB and the less affected by outliers median stands at 2.23MB. Overall, all the websites take 184.93GB of space.

CO_2 Distribution

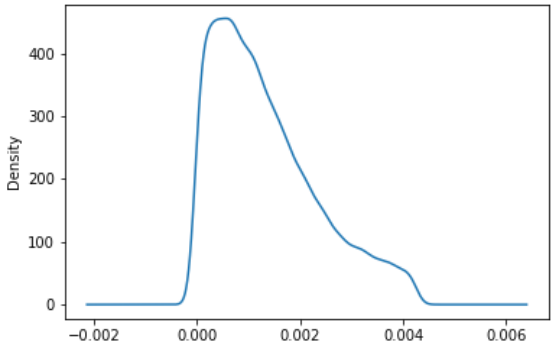
In total, the sites generate 46554 grams/25893 liters of CO_2 and on average, a single site generates 0.562 grams of CO_2 per first page load. The minimum amount registered is 0.000042 but the largest one is much bigger, at 76 grams per load. The mean and median numbers here are 0.9304637 grams/0.5175239 liters and 0.5629283 grams/0.3131007 liters. As a quick comparison we can use the estimated average per vehicle per year, which stands at 4600 kilograms (Greenhouse Gas Emissions from a Typical Passenger Vehicle, n.d.). We can see that the dataset's footprint is as much of that of 10 vehicles. The following two density plots show the distribution without the presence of outliers on the left and with them on the right.

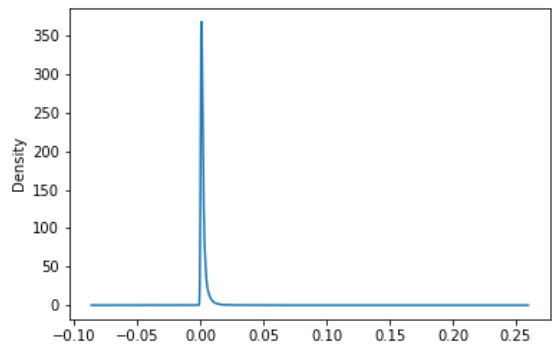




Energy

All sites use up 105 KWg of electricity, the biggest one uses 0.17 KWG and the smallest one 0.00000009568445 KWG. The mean and median figures are 0.002105122 and 0.001273593 respectively. That is a 99 and 99.993% difference when compared to the one using up the most. As a comparison, the average monthly US household consumption is 893 kilowatt hours per month (EIA, n.d.). This measurement and the kilowatt hour per gigabyte one used throughout this thesis are not strictly equivalent but even then we can clearly see a rough estimate of the usage as compared to real world usage. This, and the dioxide usage will be discussed in more detail in the upcoming sections. The following two density plots show the distribution without the presence of outliers on the left and with them on the right.





Green Hosting

Here we see that there are less websites classified as using green hosting than regular grid. The green websites take up 80.8GB of space against 104.06GB for the standard ones and the other three statistics follow a similar pattern: 20 kilograms of CO_2 are generated for all energy efficient sites against 26 kilograms for the ones labelled "unknown".

Green Hosting	Bytes	Statistics: Energy (KWG)	Statistics: Grams of CO_2 (Grid)	Statistics: Liters of CO_2 (Grid)
Unknown	104068160248	59.272054	26198.247878	14571.46547
True	80863090299	46.055599	20356.574757	11322.32688

Outliers

One of the first things noticed after the inspection of the dataset in both by histograms and manually was the presence of outliers in the data. Outliers are defined in differing ways in statistical literature. Hawkins (1980) describes an outlier as "an observation which deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism". Furthermore, Grubbs (1969) states them as "An outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs". The general meaning behind the different definitions though is that an outlier is a point of data located much farther from the average than most of the dataset.

Outliers can have a negative impact on a quantitative analysis for a variety of reasons, one of them being that they can influence important estimates in a very negative way. Essentially, there are two major reasons for the existence of errors in a dataset: human error (wrongly inputted data) or technical error (miscalculation by the system).

Outliers come in different types. In data mining, they can be global, collective, contextual and in general statistics they can also be univariate and multivariate. Global outliers are those that "all outside the normal range for an entire dataset" (Alghushairy et al., 2020) and univariate outliers are defined by Tabachinck & Fidell as "a case with an extreme value that falls outside the expected population values for a single variable" (2018). All of the outliers which are discussed in this section fall in those two categories.

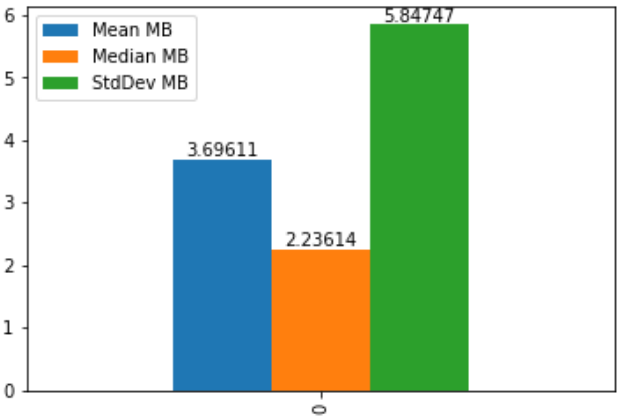


Figure 2: Average, Mean and Standard Deviation for "Bytes"

Understanding the nature of our outliers is important to the nature and validity of the data. What made them occur? Was it human error or a technical one, and what do they say about the websites?

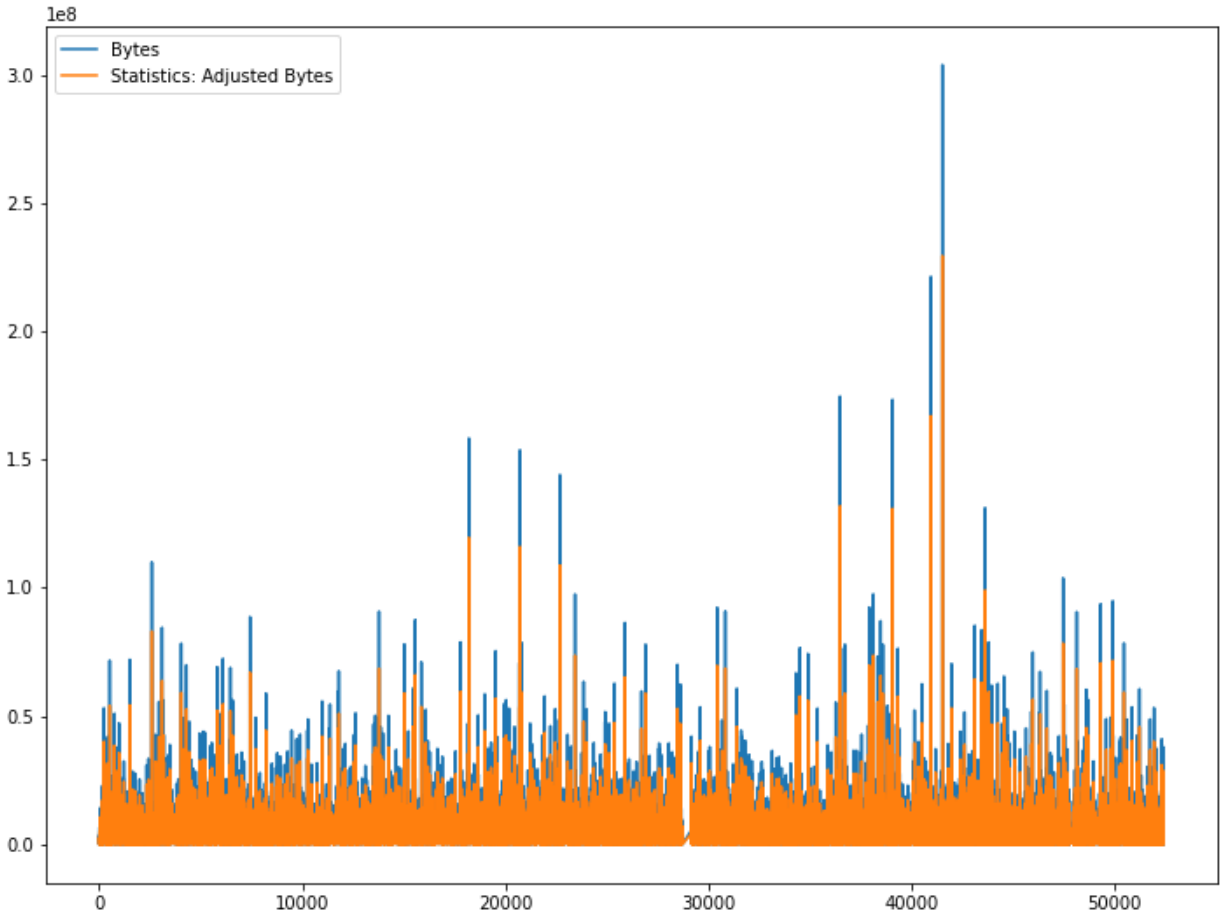
At first look at the dataset, we notice a large difference in website sizes. Looking at the "Bytes" and "Adjusted Bytes" columns we see that although the average website size in "Bytes" stands at 3.69MB, the mean is only 2.23MB, a clear sign of the distribution being skewed. Looking at the top 10 and bottom 10 values in "Bytes" shows us just how large the difference is, especially when compared to the already mentioned average.

Top 10 Bytes	Bottom 10 Bytes
304084175	168
221431538	170
174746287	185
173488420	200
158429770	207
153772730	209
144149539	209
131234648	218
110049081	227
103846179	230

The largest recorded website is 304084175 bytes (304MB) and the smallest only 168 bytes. From the top 10 we see that each website drops off in size from the previous one by anywhere between 5 and more than 80 megabytes and the 10th is "only" 110MB. The bottom ten looks very different with all the websites weighing at less than 1/5th of a megabyte. The outliers deviate by a large amount but defining what is an outlier is difficult to do and it depends entirely on what the threshold in "Bytes" should be for that. If we look at the 25th, 50th and 75th percentile we can see that the 75th is measured at 4.23mb, only slightly higher than the average, yet the maximum is 300mb. Clearly the percentage of outliers is large. If we take an outlier to be any site larger than 5mb then we are left with 40047 websites, a loss of 20%. If the number is changed to 10mb we have 46692, a loss decrease of 13%, down to 7% overall and if we simply take the dataset average of 3.69mb then the loss increases with almost 10%, up to almost 30 because we are left with only 35153 domains.

In the end, the decision on which sites are outliers is purely subjective.

The whole picture can be seen in the following plot:



After performing tests on a random selection of the outliers a few different patterns are easily observed, but none of them can be called the sole reason for the difference in estimates. In some cases, the cause was simple: the website was offline; thus, the API did not record anything beyond a generic browser response. Others were a blank page, sometimes with a few lines of text, or were not indexable due to the several reasons listed by Website Carbon which have been mentioned earlier.

All of these relate to the bottom ten. For the top ten though, the reason was much more surprising and interesting. It had to do mostly with the page contents not being optimized at all.

Before anything, it bears remembering that some websites are more demanding by nature. Streaming services of course often download videos locally, but news sites on the other hand have a lot of dynamic content which gets updated daily or even hourly.

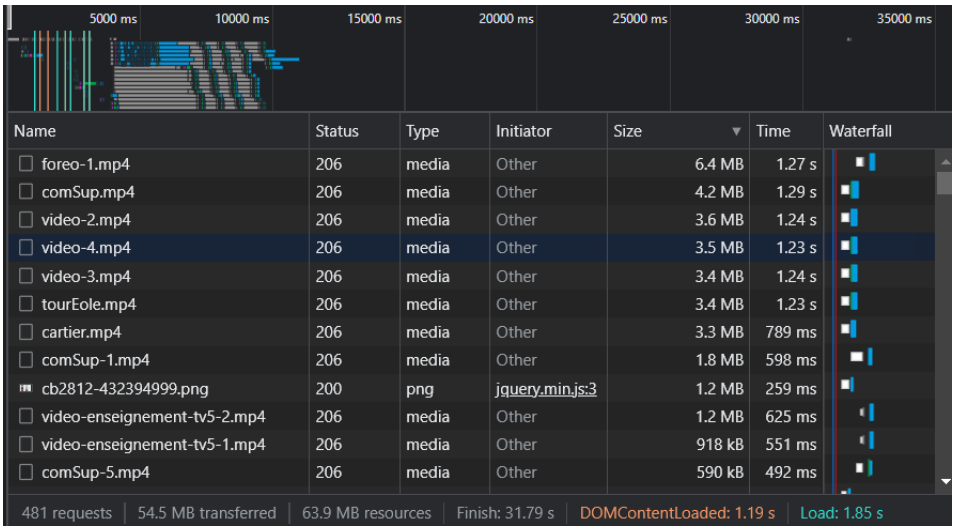


Figure 3: An example of the largest files

The largest website mentioned previously, and one that will serve as a general example is lematin.ma, the online version of Le Matin, a Moroccan daily newspaper. This site, tested initially on 15/05/2022 was estimated to be 304MB, as previously stated, but on further testing it returned vastly differing sizes, ranging anywhere from the original estimate, to 11.4MB, as of 10/06/2022. Examining the website with Google Chrome's Developer Tools, caching disabled, no advertisement blocking, and screen resolution forced to 1080p within the browser (to prevent the mobile version from appearing) was done on several different days and it showed that the website is filled with heavily unoptimized images and videos, most of them coming from article previews, automatically scrolling sections and ads. For example, on 27/05/2022 there were two identical ad videos hosted on the website each of them being 84.4MB. On 06/10 again, minutes after the previously mentioned test, the website increased from 11.4 to 169 megabytes after an automatically triggered refresh and multiple videos ranging from 5 to 73mb were loaded. Upon further refreshes, none of which were triggered from my side, the size decreased to a "mere" 117MB and then finally at 54MB. Due to the dynamically changing nature of the website, some of the biggest files previously noted were only loaded on some occasions which is the main reason the amount of data transferred differed so much.

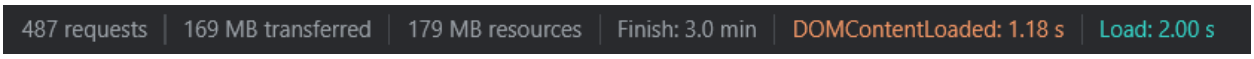


Figure 4: 06/10/2022, 16:53, LeMatin.ma

Figure 5: 06/10/2022, 16:59, LeMatin.ma

The same essential pattern shows up on many of the large websites in the dataset. Ittefaq.com.bd, a Bangladeshi newspaper exhibited similar size changes, it was initially recorded as being 158MB in both Website Carbon and Chrome Developer Tools, but on 10/06 it only transferred 11.9MB.

The other sites exhibiting those patterns were either sites with heavy graphics like warnerbrosgames.com[16] where the heavy content was in the shape of game advertisement videos or adult content streaming services some of which were loading the videos automatically and one particular example even hosted an entire visual novel game on the home page. Going back to the Warner Bros Games example, that website was initially measured at 174mb by Website Carbon. On 18/06 though it stood at 52.1mb as measured by Chrome, with 49.4 of them being all video files. That is 94% of the site's weight contributed to a few files each of which took seconds to load.

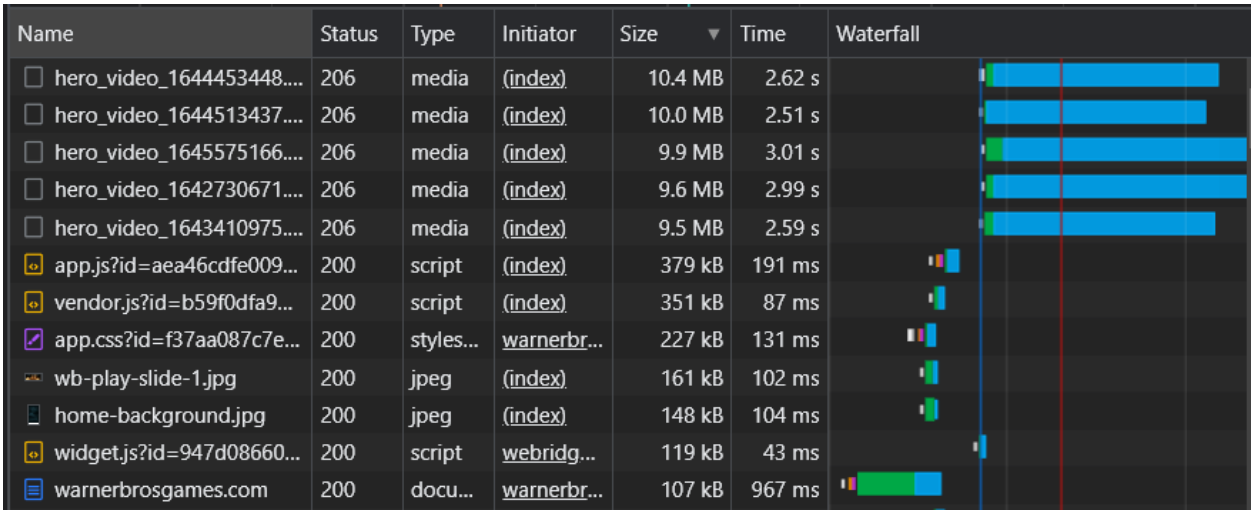


Figure 6: 18/06/2022, https://warnerbrosgames.com/

[16] https://www.warnerbrosgames.com/

For both of these examples there were inconsistencies found with regards to the sizes that Developer Tools and Website Carbon were both reporting. Lematin was mostly measured at around 300mb in WC and 170mb in Chrome, and a similar type of inconsistency was found across other websites. While researching this it was found that in the estimates given by Chrome there were often media (high-resolution videos and images) which were returning a "206 Partial Content" response code which is defined by Mozilla as "request has succeeded and the body contains the requested ranges of data, as described in the Range header of the request" (MDN, n.d.). What this means is that the data is recognized by the website but not fully loaded unless it is needed.


```
{
  "url": "https://lematin.ma/",
  "green": true,
  "bytes": 311880836,
  "cleanerThan": 0,
  "statistics": {
    "adjustedBytes": 235470031.18,
    "energy": 0.17763183010350914,
    "co2": {
      "grid": {
        "grams": 78.51326890575103,
        "litres": 43.669080165378716
      },
      "renewable": {
        "grams": 68.0685172956647,
        "litres": 37.859709319848704
      }
    }
  },
  "timestamp": 1656158120
}
```


Figure 7: Lematin.ma, as measured on 25/06/2022

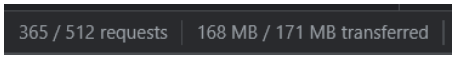


Figure 8: The same website, 25/06/2022, Google Chrome

The reason why Website Carbon reports a much higher number is rooted in the way it works. After consulting with two of the developers from Wholegrain Digital I was told that the API uses Google's PageSpeeds Insights. There, the total size was identical to the one reported by WC. Some of the videos found hosted on Lematin's servers were listed on the report generated by PageSpeeds. Those were found to be the same ones that had a "206 Partial Content" as mentioned in the previous paragraph. They are used in dynamic content heavy sections such as scrolling news sections which are triggered by a mouse click. The same type of concept can repeat in other websites as well, either measured or not measured in the dataset used in the thesis.

Other websites tested through the API and Chrome at the same time gave much more comparable results. Youradio.cz, for example, was tested on 25/06/2022 and gave very similar results in both tools, 5.2mb of data were both detected there. The same repeated for other websites. What I found out is that the behavior exhibited by the API (and PageSpeeds as well) is comparable to using Linux's 'wget' with a high level of recursiveness enabled which parses links found on the website and saves them even if the content is not directly hosted on the server.

Such type of content is not guaranteed to be loaded on a first visit of a page, but it is nevertheless highly possible to be triggered by regular user behavior and is counted here towards a website's size.

All of these tests were generally performed at random intervals, and for one reason: to determine whether retesting would be needed to verify the correctness of the data. One important fact about the nature of the outliers has been verified by this, *the data is not a product of human or technical error, it truly represents a website's state at the time of testing and gives clear examples to the importance of proper web development done with respect to standards and quality*.


```
{
  "url": "https://www.youradio.cz/",
  "green": "unknown",
  "bytes": 5302843,
  "cleanerThan": 0.18,
  "statistics": {
    "adjustedBytes": 4003646.465,
    "energy": 0.0030202359302435072,
    "co2": {
      "grid": {
        "grams": 1.3349442811676302,
        "litres": 0.7424960091854358
      },
      "renewable": {
        "grams": 1.157354408469312,
        "litres": 0.6437205219906311
      }
    }
  },
  "timestamp": 1656156138
```


Figure 9: YouRadio.cz, as measured on 25/06/2022

The results that are used in this section though could be considered slightly subjective as some estimates are evidently slightly inflated, as far as a first load of a webpage is concerned. What will be shown in the next several

sections should not be considered not to be perfectly accurate, but it is nevertheless proof considering that browsing habits are not limited only to a homepage.

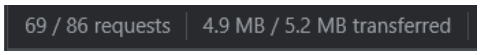


Figure 10: The same website, in Chrome

Top and Bottom 1000

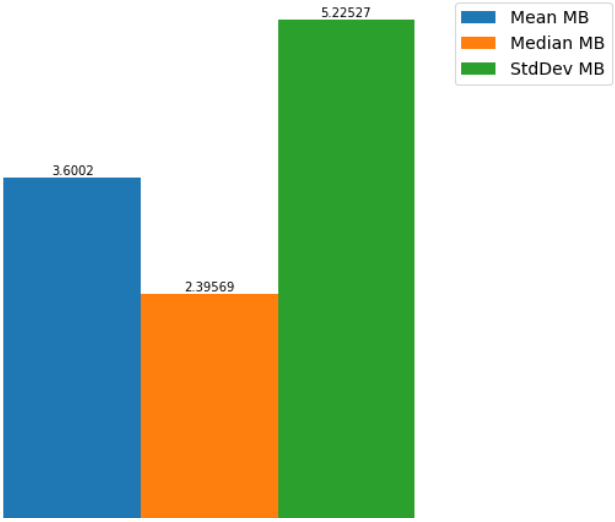


Figure 11: Top 1000, Mean, Median, St Dev in 'Bytes'

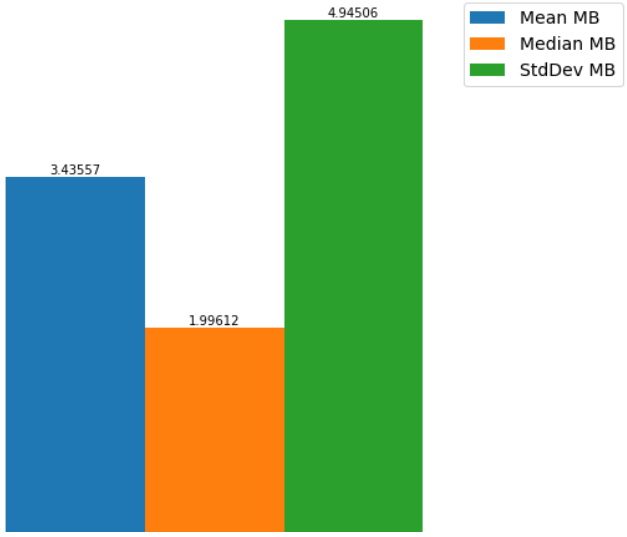


Figure 12: Bottom 1000, Mean, Median, St Dev in 'Bytes'

Here we observe similar numbers to those of the entire dataset. The top 1000's average sizes are only slightly lower than the entire dataset's and the same goes for the bottom. Overall, the differences are negligible with the top 1000 being only slightly lower (0.09mb) in terms of size than the 3.69 mb originally measured and the bottom is 0.14mb less. Those differences can be explained as small deviations caused by the presence or lack thereof of one of the previously detected outliers in this slice of the data, and that can be further seen in the bigger difference in the median sizes where there is a 0.4mb disparity. As far as the CO_2 output goes, we also see a similar picture, as the averages stand at 0.90 and 0.86. The size numbers are very comparable to those reported by (Everts, 2017) in 2017 and the small difference with the 2022 data is caused by either natural increases in file sizes year-by-year or by the already discussed outliers.

Type	Top, MB.	Bottom, MB.	Top, Energy	Bottom, E.	Top, Grams	Bottom, G.
Count	3600.20	3435.57	2.05	1.96	906.32	864.88
Mean	3.60	3.43	0.002050	0.002	0.906	0.865
Median	2.40	2.00	0.001364	0.001	0.603	0.503

Hosting Type

As previously noted, we have used two different definitions for the hosting type. "True" signifies that the site is powered by a data center using renewable energy and "unknown" means that the energy type could not be confirmed. It is worth mentioning again that the hosting status is strictly determined by whether the domain is registered with the Green Web Foundation, and it is entirely possible that some of the "unknown" labeled sites could also be renewable. That is only confirmable on a case-by-case basis though and is beyond the scope of this paper. The main takeaway here should be that the "unknown" values might be slightly skewed. In this case though we will take them at face value.

Unknown	25708
True	24326

For the entire set we have approximately the same number of sites for both types, 25708 unknowns and 24326 greens, the difference is only 6%. What is interesting though is that the difference is much larger in the other statistics. Referring back to table 1 we notice a much larger 23% difference in website size, energy usage and also CO_2 generation. On average, a site using renewables uses 3.32mb of space and those that do not use 4.04mb which means that either there simply are more outliers in the second group or that developers who have built overly large sites are not particularly concerned with whether they are carbon neutral or not.

To verify that, we first split the dataset in two, one half only containing True and the other only Unknown. The distribution between green and regular usage in the top websites is roughly equal, with many often-used websites contained in both the first and second groups. Some examples include Google, YouTube, Facebook, Netflix, and Instagram being hosted renewable and Microsoft, Twitter, LinkedIn, and Wikipedia using regular energy.

What also is curious that there are indeed many more large sites in "unknown", but on average they are actually slightly smaller in size than the renewable ones. That can be seen by further modifying the two split datasets to be arranged by descending and seeing an overview of the sites in both statistics. If we define a large site to be one over 100mb then we see that True and unknown both contain only 5 sites, and the difference mostly comes from the first two sites which are much larger than the others in size. By filtering down to those larger than 50mb, the picture starts to change, the sizes are almost the same in all rows. But once we filter the sets to only include anything over 25, 15 and 10mb we start seeing some large differences. The 25mb column already has a large difference in size, caused by the larger number of sites in "unknown", even if those are generally smaller than the trues and this continues into the 15 and 10mb columns too. Although the average size stays slightly lower in weight in "unknown", there are 28% more of them and that bloats the overall weight. All of this is shown in detail in the table below.

Type	>= 100mb	>= 50mb	>= 25mb	>= 15mb	>= 10mb
True Bytes	897840743	4082672748	9812265588	18259742599	27146551273
Unknown B.	777391624	4024409192	11584901699	22420267851	35320806777
True Count	5	53	222	672	1407
Unknown C.	5	53	285	867	1935
True, per site	179,568,148.6	77,031,561.28	44,199,394.54	27,172,236.01	19,293,924.14
Unknown, p.s.	155,478,324.8	75,932,248.90	40,648,777.89	25,859,593.83	18,253,646.91

The takeaway here is that websites using regular energy are quite a bit more likely (28%) to be demanding and to have a larger emission footprint.

Per Domain

This next section will look at the trends across the different domain types on the internet and will see if there is a difference between them as far as our original statistics go.

Original Domains

The seven "original top-level domains" were created in the 1980s to cover the needs of the first websites on the internet. They are: .com, .edu, .gov, .int, .mil, .net, and .org (ICANN, n.d.).

TLD	Amount
.com	27884
.org	3873
.net	2066
.edu	921
.gov	387
.int	24
.mil	5

Between these seven we have 35161 websites for a total of 130604115266 bytes/130gb. The distribution is shown in the table on the left.

Applying the same type of analysis as before we see that the averages and medians are not too different than the entire dataset as they stand at 3.71/2.25mb.

When comparing the domains directly we see a slightly different picture.

Type	.com	.org	.net	.edu	.gov	.int	.mil
MB's	103289.11	13521.25	5951.96	6342.36	1351.43	111.69	36.30
Mean MB.	3.70	3.49	2.88	6.89	3.49	4.65	7.26
Median MB.	2.26	2.13	1.44	4.84	2.40	3.29	3.63
Energy	58.83	7.70	3.38	3.61	0.77	0.06	0.02
Mean E.	0.002110	0.001988	0.001641	0.003922	0.001989	0.002651	0.004135
Median E.	0.001290	0.001216	0.000825	0.002757	0.001366	0.001873	0.002071
Grams	26002.13	3403.85	1498.35	1596.63	340.21	28.12	9.14
Mean G.	0.93	0.88	0.72	1.73	0.88	1.17	1.83
Median G.	0.57	0.54	0.36	1.22	0.60	0.83	0.92

What this tells us is that most TLD's follow the set averages, with the exceptions being .edu, .int and .mil. Those four are smaller than the entirety of .com but they do contain a significant number of weighty sites.

For the energy consumption values and grams, we see repetitions comparable to the ones in "Bytes". "Edu", "Int and "Mil" are expectedly higher than the rest. Those domains are less than 2% of the entire dataset though and they do not have a major impact on the overall footprint. The full overview can be seen in the table above.

Regional Domains

Here follows a brief analysis on the regional domains found in the dataset. Those are all country specific domains that could be detected with regular expressions (e.g., ".co.uk" for the United Kingdom and ".bg"

for Bulgaria"). They are not grouped by the location of the hosting server or the actual origin of the website as the latter is impossible to determine for all websites and the former can be considered meaningless as many websites nowadays are hosted in countries other than the one they are targeted for. This section analyses 10,180 domains, or 20.3% of the dataset.

Region	EU + UK	North America	South America	Asia	Middle East	Oceania	Africa
MBs	14441.05	2466.47	3473.42	16320.58	2234.84	1474.78	1346.97
Mean MB	3.27	3.64	4.11	4.71	4.56	3.65	4.60
Median MB	2.23	2.42	2.69	2.99	2.83	2.97	2.16
Energy (KWG)	8.22	1.39	1.98	9.29	1.27	0.84	0.76
Mean E.	0.001862	0.002072	0.002338	0.002681	0.002597	0.002079	0.002618
Median E.	0.001268	0.001	0.001531	0.001705	0.001612	0.001693	0.001231
Grams	3635.40	615.88	874.40	4108.56	562.60	371.26	339.089
Mean G.	0.82	0.92	1.03	1.18	1.15	0.92	1.15
Median G.	0.56	0.61	0.68	0.75	0.71	0.75	0.54

Region	Amount
EU + UK	4416
North America	671
South America	846
Asia	3466
Middle East	490
Oceania	404
Africa	293
Total	10586

The two biggest groups here are the European and Asian domains. They represent 74.4% of the 10,586 regional websites and as such also generate the most dioxide. The European domains are smallest in size and the Asian + Oceanic ones are 26% larger. All regions exhibit large disparities between the mean and medians of all three categories of data (MBs, Energy, Grams) which of course indicates the presence of significantly sized outliers but the medians for energy and grams show that the African domains are the most efficient with 0.54g, followed closely by the European ones with 0.56.

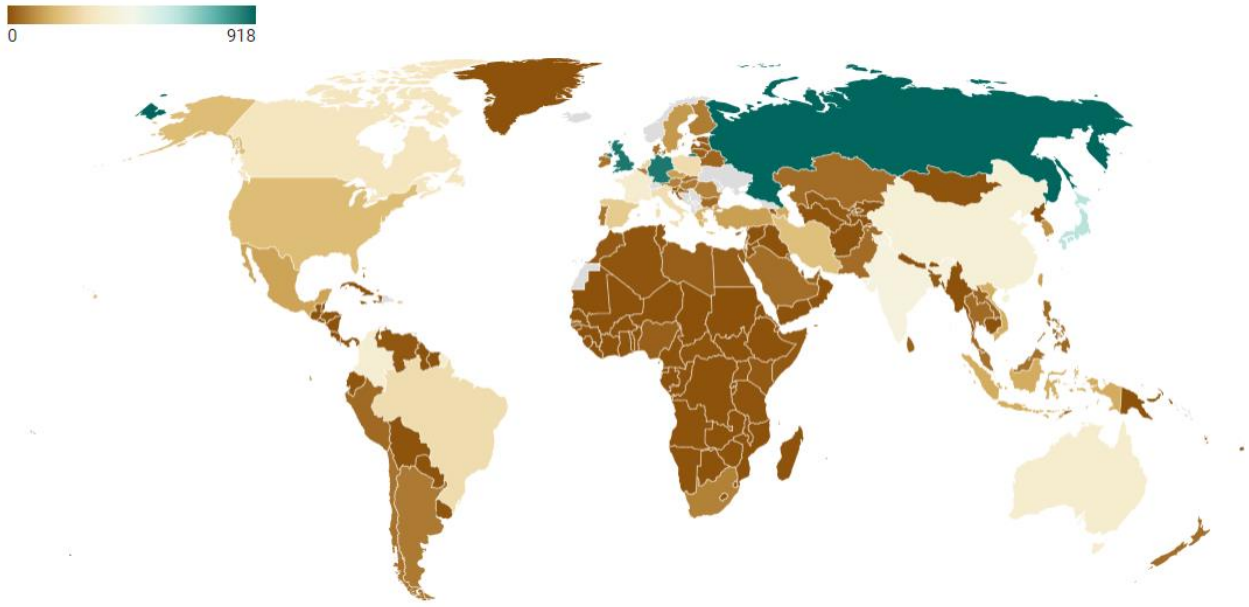


Figure 13: Top-level domains per country, choropleth map

6. Discussion & Results

In order to truly understand this data though and see the impact a website has on the environment it is necessary to know how it compares to real world examples. To do so the data will be now compared against measurements sourced from the International Energy Agency. Then we will look at a small sample from the dataset and discuss possible improvements for web development.

The average emissions per capita vary greatly. IEA measures the footprint for a United States citizen to have been 14.4 tCO_2 (IEA, n.d.). This number is very different than the average for Europe, which is 5.5 tCO_2 or the one for Africa (1.0 tCO_2). For this reason, the world average will be used here – 4.4 tons per person, per year, or 12kg per day.

The collected data had several limitations as stated already. The major one is that the only pages tested were homepages, some of which can be login pages depending on the website. It is impossible to measure everything in the most accurate manner and because of that the data will have to be taken at face value to be able to discuss it.

We saw many different measurements in the analysis so far. As a whole, the data variability is high, with highs more than 1.8 million times larger than the lowest recorded value but depending on what the determined threshold for defining outlying data is, the variability stabilizes with a significant amount as the standard deviation can go from anywhere between 5.84 to 2.16 depending on which of the three outlier thresholds shown in “Outliers” are used.

That is to say though, those small in the scope of the thesis changes can have a very significant impact at a large scale. As we saw before, the average site size generates 0.93 grams of dioxide per page click and uses 0.002105 KWG of electricity. Loading all 50 thousand at once is equivalent to 46.5 kilograms, almost as much a single person does for 4 days. But that is only the average and with the already seen regional and per-domain differences this presents a real and significant danger.

A website can have anywhere between 1 and billions of clicks per month and 20 of the most popular websites on the internet alone generate over 100 billion visits per month (Statista, 2022). Going by these statistics alone, the internet's impact is clearly massive as those twenty domains alone amount to over 93,000 tons per month, if every click was viewed as a first visit and we only used the previously mentioned average.

Those numbers can change significantly. If we apply the generally seen 25% reduction in data transfer by using caching on repeated visits then we get at a minimum 69,750 tons p/m or 837,000 per year and 105.33 kilowatts per gigabyte. If the 4.4 tCO_2 per person, per year mentioned earlier are accurate then this means that the twenty most popular websites emit at least as much as 190,000 people, roughly the same size as the city of Breda, NL and use more than 130 billion kilowatts per gigabyte.

Another possible way to look at this would be to utilize the regional measurements and parse the information in a more fine-grained way. As stated before, the small changes per region are in reality significant at a large scale. The carbon footprint per continent ranged anywhere between 0.82 and 1.26 grams of CO_2 per click, which in theory could lead to a 11% decrease/33% increase in emissions when visiting those sites, without accounting for caching.

Nevertheless, that ignores the bigger problem: the difficult to measure user behavior.

There are at least 1.88 billion websites in the world (Armstrong, 2021) and if each of these were opened just once per day then we'd be contributing negatively to the environment with at least 1748 tons of dioxide each day (again, if the estimated average is used for these calculations), more than 52 thousand each month and 638,000 each year (478,000 if adjusted for caching). If a user then visits 50 pages per day we might be looking at least at 32 million tons per year (23.9 million adjusted) which is as much as the population of a country of 7 million people could emit in a year by themselves.

Unfortunately, there is no realistic way to know that with precision as user behavior across the planet can be affected by many different reasons, ranging from the time of day to the type of the website or by local events or unpredictable natural phenomena.

7. Conclusion

The research question was defined previously as:

What is the current state of energy consumption of Web sites?

While many web pages are often well optimized for speed and weight, it is not all of them.

This thesis presented a simplified look at the state of some of the most popular websites on the web, an often-overlooked part of the existence of the internet. The domains were analyzed both in terms of how weighty and harmful they can be but also in how much of an impact they have on the electricity grid.

We found that the web can be unpredictable. Depending on the criterion used to determine large websites we can define anywhere between 7 and 30% of the analyzed data to be an outlier, and a significant amount of those excluded websites are multiple times larger than the norm.

Generally, large differences in the footprint can be found between different regions, domain types and also between different parts of the dataset. The type of hosting is a major sign that a website can be

sustainably built, as the Green labeled datacenters were hosting domains which were around 20% smaller on average than the rest.

Although in multiple sections the differences could be considered small at first read, they carry a large impact when scaled up. A gap of 5 to 20% or more can be considered unimportant for a single page load but it is of a much larger importance when that percentage can amount to thousands of tons of dioxide per day. As a webpage's size increases, so does its energy usage, and an unoptimized page can easily lead to unneeded kilowatts of electricity that could have been used for a more important reason.

The emissions left on the planet accumulate daily and the only way to counter that is to work actively on reducing them. The slice of the web that was analyzed here is in a far from perfect state. Nearly a fifth of the data was found to be overly large and as of today there is no entity or organization that deals with actively monitoring and controlling website sizes, there are only guidelines and recommendations. Because of that the responsibility here lies on web developers entirely.

The internet is of course much bigger than this analysis and the dataset that was examined is only a minor part of the whole picture, in reality it can be safely assumed that the real numbers are much higher than what was discussed here. The actual impact of the web is unfortunately impossible to measure within the time constraints given, but this is still a starting point for any additional and much more expansive analyses on the current state of the internet.

8. Reflection

The process of writing this thesis is described below:

I started working on the thesis on 14/04/2022.

Initially, the idea was to use as many of the tools available on Green Web Foundation's Awesome Green Software list[17], along with Selenium[18], a web project combining several different tools and libraries for the purposes of automation and web-scraping. Alongside that, I also began researching carbon emissions in both general aspect and with a focus on IT and the web. The overall goal was to analyze the entire Tranco ranking list of 1 million websites.

Several days later, after testing all of the GWF tools, I realized that almost all of them were unfortunately outdated, most likely inaccurate, impossible to use for the scope of the project or all of these at the same time. This is described in more detail in 7.3. One thing became clear, either the entire goal and scope had to be scaled down drastically, or I had to use the only one that could really fit with the thesis: Website Carbon.

At that point in time, the idea was still to scrape all of the data. I built a basic Python scraper with Selenium and started testing it with the API. Each scrape took approximately 10 to 20 seconds to finish, which meant that I'd need at the minimum 234 days to parse the entire ranking list. The reason for that was that a scraper essentially simulates the actions a person can take on a website, which meant waiting for page loads, server slowdowns and so on. As the API took differing amounts of time to process each site, there was no viable way to force sub-10 second waiting times for scraping and the server was slowing down

[17] https://github.com/Green-Software-Foundation/awesome-green-software#web
[18] https://www.selenium.dev/

with time. Additionally, I also tried the scraper on one of the other faster sites, Kastor.green but a scrape there took even longer than that.

Around this time, I accidentally found a (then) unpublished API description page for Website Carbon, with the public API endpoint mentioned on it and guidelines on how to use it. Now I could receive JSON formatted responses in return.

I wrote a HTTP GET requests code, in Python again and with the Requests library, and tested the API. The results were much faster, around 15 minutes per 200 parses and the API was also providing additional data that was not used on the main Website Carbon page. There were several issues with this approach though. First, the API was not as reliable as I had hoped, many websites were not returning any data and my success rate was only around 16% as for every 5000 requests I was receiving only 800 or so. After testing this I realized that my queries were overloading the server and that some domains were simply impossible to test, due to the reasons already mentioned in "Errors and limitations". I adjusted the number of requests I was sending out and the response rate improved immediately (from 16 to 74%). The other issue will be discussed in the next paragraph.

After handling the response rate, I wanted to improve the speed of parsing. This is when I found aiohttp and asyncio and I rewrote the code with concurrency in mind. After rewriting everything, I managed to increase the overall speed and was now able to do 200 parses in 3-5 minutes. This, unfortunately escalated an issue I was also encountering with the non-concurrent method at first: some sites, mostly Cloudflare based ones, were returning HTML error pages instead of the JSON I needed which made the program crash. With Requests that was very easy to handle with proper exceptions handling, but that was difficult with asyncio. The main reasons for that were my lack of experience with concurrency programming and the way methods are handled when utilizing it. Contrary to sequential code, concurrency executes methods which are usually reached last multiple times while some of the older ones are still being ran at the same time. To explain this more succinctly, here's an example: the get() method in get_requester.py can be executed and return data hundreds of times while an older method like main() can append more tasks to the 'tasks' variable. The first thing I did was to handle it with exceptions like last time, but the exception was not being reached because of the methods being executed out of order. Due to my lack of knowledge on this topic and the time constraint I had I decided to ignore the problem and continue gathering data by splitting it in smaller chunks, 200 at a time, and continue. This was unfortunately taking a little longer due to me having to restart the code often, but it was still several times faster than the sequential approach. In the end I settled on the 50,000 usable websites and focused on writing the thesis.

During the writing of the thesis there were several other limitations encountered which led to changes in the scope of the final deliverable. They are:

8.1. Lack of temporal data

The API used for gathering the data only provided measurements for the state of a website at the exact time the test was being established. Access to data older than what was gathered would have allowed for an overview of the progression of website sizes through the years. Ideally, such data would have existed for at least the last couple of years if not more. They would have at minimum included website sizes which would have helped calculate a very rough estimate for previous energy usage (ignoring technology efficiency improvements), but the API is too new for this and nobody before has done large scale measurements on a regular basis.

8.2. Lack of bandwidth

The initial suggestion was for the entire Tranco list to be processed. Unfortunately, that was not possible due to the earlier mentioned limitations with the API. The 2000 possible website measurements per day were too limiting to process the entire dataset as it would have taken at minimum 500 days to collect all of the data, excluding any other possible issues like site outages. From early on a relatively arbitrary amount of 50,000 was decided as it was neither too little to give a decent overview and neither too large to take up too much time to collect.

8.3. The tools available were not functional

Originally this paper would have included a wider array of measurement tools from the list mentioned at the beginning of this chapter. Many of them were either not functional, paid, too slow to use in this limited timeframe or were simply giving very different measurements than the other tools. Below follows an overview of each and the reason why they were not used:

Tool	Reason
Carbonalyser[19]	Last updated in January 2020. In theory useful as it can measure the network traffic as it is happening but the numbers are too different to be compared directly with the main analysis in the thesis (e.g., loading VU Amsterdam's homepage measures the carbon output at 1g, whereas the other tools in this table give measurements 2.5-8 times higher than that).
Carbon Footprint of Sending Data[20]	An interesting calculator which could give an outlook at a site's footprint but the numbers that we get out of it are too different than any of the other tools (e.g., a 9mb website is estimated to generate 810kg of CO2 per month but Eco Grader and Website Carbon estimate that to be 272.03 and 235kg's respectively). There is no way to truly know which one of the three is accurate.
Clickclean[21]	It is very outdated (from 2017) and only focused on mobile apps.
CO2.js[22]	Useful, but its functionality is already implemented in Website Carbon.
EcoGrader[23]	Similar to Website Carbon but the only usable statistic here is the grams of CO2 that a website generates which is already provided by WC. Also, the two websites use different methodologies for the CO2 statistic which leads to different results. It is not clear which one is the most accurate. All other statistics appear interesting at first sight but after further inspection we can see that they are either vague (only expressed in an abstract 1-100 scale) or concerned with usability than actual environmental impact.
EcoMeter[24]	Does not work, the results never appear.
GreenFrame[25]	I had a genuine interest in this one, but it is built for webapps currently in development, not finished and only for already hosted websites.

[19] https://theshiftproject.org/en/carbonalyser-browser-extension/
[20] https://observablehq.com/@mrchrisadams/carbon-footprint-of-sending-data-around
[21] http://www.clickclean.org/
[22] https://github.com/thegreenwebfoundation/co2.js/
[23] https://ecograder.com/
[24] http://ecometer.org/
[25] https://greenframe.io/

Mobile Efficiency Index[26]	Results are only receivable through email, take 10 to 20 minutes to be sent out and same as most of the tools here, the measurements are too different.
Kastor.green[27]	One of the few semi-useful sites from the list, it was considered to be used in the Discussion section until the end but there seems to be too large of a difference in some of the estimates. It finds extra .js files that Chrome does not catch and yet the total size is vastly underestimated in most cases. Case in point: nu.nl is measured to be 1.889kb by Kastor but 2.7mb in Chrome. Additionally, many of the recommendations in the audit generated by the site are too vague (e.g., "Reduce unused JavaScript") to be used here. Some are also too general ("Too many [content] requests") which in this case shouldn't apply to a news website.
WeDeex[28]	A Chrome/Firefox extension that does not support either large-scale analysis or exporting the data in a machine-readable way.

References

206 Partial Content - HTTP | MDN. (n.d.). Mozilla Web Docs. Retrieved June 30, 2022, from https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/206#:%7E:text=The%20HTTP%20206%20Partial%20Content%20success%20status%20response%20code%20indicates%20that%20the%20request%20has%20succeeded%20and%20the%20body%20contains%20the%20requested%20ranges%20of%20data%2C%20as%20described%20in%20the%20Range%20header%20of%20the%20request.

Alghushairy, O., Alsini, R., Soule, T., & Ma, X. (2020). A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. *Big Data and Cognitive Computing*, *5*(1). https://doi.org/10.3390/bdcc5010001

Armstrong, M. (2021, August 6). *How Many Websites Are There?* Statista Infographics. Retrieved June 25, 2022, from https://www.statista.com/chart/19058/number-of-websites-online/

Aslan, J., Mayers, K., Koomey, J. G., & France, C. (2017). Electricity Intensity of Internet Data Transmission: Untangling the Estimates. *Journal of Industrial Ecology*, *22*(4), 785–798. https://doi.org/10.1111/jiec.12630

Andrae, A. (2020). New perspectives on internet electricity use in 2030. *Engineering and Applied Science Letter*, *3*(2), 19–31. https://doi.org/10.30538/psrp-easl2020.0038

[26] http://mobile-efficiency-index.com/en/
[27] https://kastor.green/
[28] https://chrome.google.com/webstore/detail/wedeex/ojlagggckhpedblhemgjhecbggnibale

Büchel, K. (1996). System Boundaries. *Life Cycle Assessment (LCA) — Quo Vadis?*, 11–25. https://doi.org/10.1007/978-3-0348-9022-9_2

Cisco Systems, Inc. (n.d.). *Cisco Annual Internet Report (2018–2023) White Paper*. Retrieved June 19, 2022, from https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

Everts, T. (2017, August 9). *SpeedCurve | The average web page is 3MB. How much should we care?* SpeedCurve. Retrieved June 25, 2022, from https://www.speedcurve.com/blog/web-performance-page-bloat/

EPA. (n.d.). *Global Greenhouse Gas Emissions Data*. US EPA. Retrieved June 19, 2022, from https://www.epa.gov/ghgemissions/global-greenhouse-gas-emissions-data#:%7E:text=Global%20carbon%20emissions%20from%20fossil,increase%20from%201970%20to%202011

Fabbrizzi, S., Maggino, F., Marinelli, N., Menghini, S., Ricci, C., & Sacchelli, S. (2016). Sustainability and Food: A Text Analysis of the Scientific Literature. *Agriculture and Agricultural Science Procedia*, *8*, 670–679. https://doi.org/10.1016/j.aaspro.2016.02.077

Freitag, C., Berners-Lee, M., Widdicks, K., Knowles, B., Blair, G. S., & Friday, A. (2021). The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations. *Patterns*, *2*(9), 100340. https://doi.org/10.1016/j.patter.2021.100340

Frequently Asked Questions (FAQs) - U.S. Energy Information Administration (EIA). (n.d.). EIA.Gov. Retrieved July 14, 2022, from https://www.eia.gov/tools/faqs/faq.php?id=97&t=3#:%7E:text=In%202020%2C%20the%20average%20annual,about%20893%20kWh%20per%20month.

Greenhouse Gas Emissions from a Typical Passenger Vehicle. (n.d.). US EPA. Retrieved July 14, 2022, from https://www.epa.gov/greenvehicles/greenhouse-gas-emissions-typical-passenger-vehicle#:%7E:text=typical%20passenger%20vehicle%3F-,A%20typical%20passenger%20vehicle%20emits%20about%204.6%20metric%20tons%20of,8%2C887%20grams%20of%20CO2.

Ge, M., & Ross, K. (2019, September 17). *Which Countries Have Long-term Strategies to Reduce Emissions?* World Resources Institute. Retrieved June 19, 2022, from https://www.wri.org/insights/which-countries-have-long-term-strategies-reduce-emissions

Grubbs, F. E. (1969). Procedures for Detecting Outlying Observations in Samples. *Technometrics*, *11*(1), 1–21. https://doi.org/10.1080/00401706.1969.10490657

Hawkins, D. (1980). *Identification of Outliers*. Springer Publishing.

ICANN | Archives | Top-Level Domains (gTLDs). (n.d.). ICANN. Retrieved June 29, 2022, from https://archive.icann.org/en/tlds/#:%7E:text=In%20the%201980s%2C%20seven%20gTLDs%20(.com%2C%20.edu%2C%20.gov%2C%20.int%2C%20.mil%2C%20.net%2C%20and%20.org)%20were%20created.%20Domain%20names%20may%20be%20registered%20in%20three%20of%20these%20(.com%2C%20.net%2C%20and%20.org)%20without%20restriction%3B%20the%20other%20four%20have%20limited%20purposes.

IEA. (n.d.). *Data & Statistics*. Retrieved June 25, 2022, from https://www.iea.org/data-and-statistics/data-browser?country=USA&fuel=CO2%20emissions&indicator=CO2PerCap

IEA. (2014b, July). *More Data, Less Energy*. https://www.iea.org/reports/more-data-less-energy

IEA. (2021, November). *Data Centers and Data Transmission Networks*. https://www.iea.org/reports/data-centres-and-data-transmission-networks/

Krisetya, M., Lairson, L., & Mauldin, A. (n.d.). *Global Internet Map 2021* [Graph]. Global Internet Map 2021. https://global-internet-map-2021.telegeography.com/

Learn About Sustainability. (2021, December 2). US EPA. Retrieved June 19, 2022, from https://www.epa.gov/sustainability/learn-about-sustainability

Leiserowitz, A. (2019, February 15). *Climate change in the American mind: December 2018*. Yale Program on Climate Change Communication. Retrieved June 19, 2022, from https://climatecommunication.yale.edu/publications/climate-change-in-the-american-mind-december-2018/

le Pochat, V., van Goethem, T., Tajalizadehkhoob, S., Korczynski, M., & Joosen, W. (2019). Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. *Proceedings 2019 Network and Distributed System Security Symposium*. https://doi.org/10.14722/ndss.2019.23386

Lindsey, R. (2020, August 14). *Climate Change: Atmospheric Carbon Dioxide | NOAA Climate.gov*. Climate.Gov. Retrieved June 19, 2022, from https://www.climate.gov/news-features/understanding-climate/climate-change-atmospheric-carbon-dioxide#:%7E:text=Carbon%20dioxide%20concentrations%20are%20rising,people%20are%20burning%20for%20energy

Masanet, E., Shehabi, A., Lei, N., Smith, S., & Koomey, J. (2020). Recalibrating global data center energy-use estimates. *Science*, *367*(6481), 984–986. https://doi.org/10.1126/science.aba3758

nationalgridESO. (n.d.). *What is carbon intensity? | National Grid ESO*. Nationalgrideso.Com. Retrieved June 20, 2022, from https://www.nationalgrideso.com/future-energy/net-zero-explained/what-carbon-intensity

Rehman, A., Ma, H., Ozturk, I., Murshed, M., & Dagar, V. (2021). The dynamic impacts of CO2 emissions from different sources on Pakistan's economic progress: a roadmap to sustainable development. *Environment, Development and Sustainability*, *23*(12), 17857–17880. https://doi.org/10.1007/s10668-021-01418-9

Statista. (2022, March 22). *Leading websites worldwide 2021, by monthly visits*. Retrieved June 25, 2022, from https://www.statista.com/statistics/1201880/most-visited-websites-worldwide/

Statista, & Johnson, J. (2022, May). *Share of users worldwide accessing the internet in 4th quarter 2021, by device*. Statista. https://www.statista.com/statistics/1289755/internet-access-by-device-worldwide/

Tabachnick, B., & Fidell, L. (2018). *Using Multivariate Statistics* (7th ed.). Pearson.

Telefonaktiebolaget LM Ericsson. (2020, February). *A quick guide to your digital carbon footprint*. https://www.ericsson.com/4907a4/assets/local/reports-papers/consumerlab/reports/2020/ericsson-true-or-false-report-screen.pdf

Telefonaktiebolaget LM Ericsson. (2021, November). *Ericsson Mobility Report*. https://www.ericsson.com/4ad7e9/assets/local/reports-papers/mobility-report/documents/2021/ericsson-mobility-report-november-2021.pdf/

The Shift Project. (2019, March). *-LEAN ICT- TOWARDS DIGITAL SOBRIETY*. https://theshiftproject.org/wp-content/uploads/2019/03/Lean-ICT-Report_The-Shift-Project_2019.pdf/

Thiagarajan, N., Aggarwal, G., Nicoara, A., Boneh, D., & Singh, J. P. (2012). Who killed my battery?: analyzing mobile browser energy consumption. *WWW '12: Proceedings of the 21st International Conference on World Wide Web*, 41–50. https://doi.org/10.1145/2187836.2187843

United Nations. (2019, July). *Digital Economy Report 2019*. https://unctad.org/system/files/official-document/der2019_en.pdf/

World Meteorological Organization. (2021). *State of the Global Climate 2020*. https://library.wmo.int/doc_num.php?explnum_id=10618

Zhu, Y., & Reddi, V. J. (2013). High-performance and energy-efficient mobile web browsing on big/little systems. *2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)*. https://doi.org/10.1109/hpca.2013.6522303

Appendix

Appendix A (from tld.xlsx):

tld	amount	tld	amount	tld	amount	tld	amount
com	27873	qa	11	com.pa	3	net.pl	1
org	3868	am	11	gob.es	3	lima-city.de	1
net	2065	moe	11	org.br	3	farm	1
edu	921	icu	11	gov.rs	3	blue	1
ru	914	gov.hk	11	gov.ru	3	photos	1
de	879	net.au	11	gv.at	2	gob.sv	1
io	585	porn	11	panasonic	2	academy	1
co.uk	522	gov.it	10	show	2	mba	1
jp	402	network	10	health	2	honda	1
gov	387	edu.co	10	toyota	2	istanbul	1
fr	345	lu	10	edu.ec	2	limited	1
co	318	ba	10	foundation	2	go.tz	1
ca	291	world	10	ac.cy	2	gov.mo	1
it	266	life	10	tf	2	pagexl.com	1
tv	262	gob.mx	10	land	2	pr.gov.br	1
in	260	art	9	market	2	fit	1
nl	251	gov.ua	9	com.mk	2	org.pl	1
me	241	org.cn	9	gov.mn	2	fish	1
cn	239	com.ve	9	ug	2	ovh	1
pl	237	com.bd	9	edu.vn	2	style	1
info	226	reviews	9	gov.by	2	th	1
es	221	nhs.uk	9	web.id	2	gov.tt	1
com.au	218	shop	9	racing	2	earth	1
com.br	199	ac.kr	8	blog.br	2	edu.bd	1
us	181	bz	8	digital	2	wf	1
ir	175	ac.il	8	cfd	2	sv	1
co.jp	172	cloud	8	kr.ua	2	web.tr	1
xyz	155	chat	8	com.tn	2	gov.jm	1
org.uk	154	com.do	8	menu	2	gov.lv	1
cc	154	gov.ph	8	edu.mx	2	edu.do	1
gr	151	tk	8	gob.cl	2	edu.bh	1
quest	140	pub	8	org.za	2	ac.bd	1
to	128	best	7	org.il	2	domains	1

ac.uk	126	email	7	gal	2	gov.scot	1
cz	125	edu.tw	7	co.tt	2	aws	1
az	114	finance	7	eco	2	co.mz	1
se	114	tips	7	law	2	cg	1
eu	108	review	7	space	2	com.tj	1
vn	105	edu.my	7	sp.gov.br	2	com.bn	1
ch	102	com.eg	7	report	2	kg	1
online	96	hn	7	gov.ir	2	com.ag	1
ro	79	go.jp	7	ooo	2	com.bz	1
live	77	cam	7	bh	2	com.cy	1
com.cn	75	lol	7	global	2	bi	1
com.tw	75	al	7	org.tw	2	co.bw	1
com.tr	72	men	7	store	2	ad	1
no	71	gob.ar	7	gov.tw	2	bs	1
fi	70	ac.nz	7	sharp	2	com.jm	1
be	68	ac.za	7	casa	2	cd	1
com.mx	68	gc.ca	7	care	2	co.uz	1
hu	66	net.cn	7	edu.gt	2	com.om	1
biz	66	onl	6	ntt	2	sn	1
ua	66	com.gh	6	vic.gov.au	2	com.ai	1
gov.in	65	ink	6	how	2	vu	1
co.kr	64	com.kw	6	leg.br	2	mw	1
co.za	64	ag	6	com.mm	2	cv	1
at	61	plus	6	vg	2	ml	1
gov.au	60	sc	6	mv	2	dj	1
top	59	gov.pk	6	youtube	2	je	1
club	57	edu.sa	6	abb	2	bt	1
ai	57	faith	6	gl	2	co.ls	1
pro	56	rip	6	ltd	2	co.nl	1
ie	56	dz	6	com.kh	2	dm	1
com.ua	56	edu.az	6	vc	2	bj	1
pt	55	trade	6	rw	2	com.gi	1
gov.uk	54	date	6	cu	2	bf	1
ac.id	53	com.uy	6	govt.nz	2	com.fj	1
cl	53	gov.il	5	community	2	co.ck	1
app	53	edu.hk	5	codes	2	co.zm	1
dk	52	game	5	bo	2	com.af	1
news	50	gov.za	5	info.vn	2	studio	1
mx	48	gov.vn	5	co.ao	2	com.gr	1
xxx	47	website	5	com.na	2	com.bh	1
com.ar	45	eus	5	gy	2	dhl	1
site	45	cf	5	realtor	2	com.ly	1

fm	44	scot	5	education	2	com.qa	1
sk	44	mil	5	photo	2	au	1
monster	44	press	5	tt	2	in.ua	1
co.il	43	gov.gr	5	com.bo	2	pp.ua	1
id	41	museum	5	red	2	netlify.app	1
gg	40	sex	5	com.ni	2	us.org	1
bg	40	tn	5	business	2	kpmg	1
edu.cn	39	gov.ar	5	in.th	2	org.ve	1
co.in	38	aero	5	school	2	co.ve	1
one	38	pics	5	com.np	2	us.com	1
edu.au	38	zone	5	as	2	altervista.org	1
kz	38	download	5	ci	2	watch	1
br	34	video	5	lat	2	goog	1
is	34	bar	5	gold	2	co.ma	1
win	34	md	5	go.ke	2	krakow.pl	1
tw	34	com.ng	5	jobs	2	js.org	1
gov.tr	32	exchange	5	gd	2	com.sv	1
co.nz	32	gov.my	5	com.vc	2	com.lb	1
co.id	31	edu.sg	5	bradesco	2	art.pl	1
by	31	cyou	5	net.tw	2	com.et	1
com.vn	30	go.kr	5	org.tr	2	com.pr	1
today	30	host	5	wales	2	com.cu	1
la	29	edu.ar	5	org.eg	2	rich	1
hr	29	bid	5	africa	2	jo	1
fun	28	cool	4	ga	2	carrd.co	1
com.my	26	mk	4	gov.ng	2	co.cr	1
lt	25	desi	4	dog	2	com.gt	1
blog	25	bet	4	eu.com	2	va	1
lk	25	travel	4	ad.jp	2	iq	1
link	25	love	4	uk.com	2	bplaced.net	1
im	24	ps	4	limo	2	freedesktop.org	1
pw	24	gov.eg	4	tl	2	com.sb	1
int	24	edu.br	4	services	2	co.vi	1
kr	24	mg	4	sm	2	gm	1
ne.jp	24	games	4	sr	2	tg	1
ae	23	ms	4	tokyo	1	rnu.tn	1
org.au	22	fans	4	ec	1	gov.dz	1
ly	22	gov.qa	4	gob.ec	1	co.rs	1
vip	21	com.pl	4	com.al	1	stockholm	1

pe	21	jus.br	4	pr	1	edu.ph	1
rs	21	gov.ae	4	vin	1	edu.mt	1
com.co	21	ac.jp	4	cr	1	na	1
su	21	city	4	edu.uy	1	my.id	1
com.hk	21	edu.pk	4	energy	1	place	1
dev	21	tm	4	org.hk	1	or.th	1
or.jp	21	pm	4	pythonanywhere.com	1	technology	1
sg	20	rest	4	day	1	barclays	1
gov.az	20	gob.pe	4	chintai	1	mus.br	1
gov.br	20	uno	4	gov.pr	1	bot	1
ph	19	do	4	software	1	sony	1
com.pk	19	social	4	edu.hn	1	ren	1
mobi	19	tel	4	ac.fj	1	rugby	1
co.th	19	ht	4	inc	1	gdansk.pl	1
ac.in	19	com.ec	4	rent	1	nrw	1
guru	19	bio	4	om	1	poa.br	1
cat	19	eg	4	edu.it	1	mp	1
lv	19	tools	4	legal	1	gov.do	1
name	18	mn	4	et	1	iki.fi	1
science	18	co.tz	4	supply	1	tj	1
edu.tr	17	gov.kz	4	brussels	1	coop	1
ng	17	ar	4	ac.ae	1	org.qa	1
ws	17	fo	4	nagoya	1	webcam	1
ac.th	16	org.in	4	press.ma	1	lombardia.it	1
gov.co	16	cash	4	go.ug	1	org.mt	1
tube	16	edu.in	4	group	1	duckdns.org	1
ee	15	company	4	marketing	1	ao	1
go.id	15	page	4	com.jo	1	nyc	1
google	15	mu	3	net.vn	1	me.uk	1
gov.sg	15	go.th	3	photography	1	bnpparibas	1
hk	15	sbs	3	church	1	de.com	1
uk	15	edu.kz	3	leclerc	1	ac.ke	1
nic.in	15	canon	3	kn	1	td	1
ac.ir	15	edu.pe	3	direct	1	london	1
stream	15	edu.jo	3	neustar	1	fyi	1
gov.pl	15	org.mx	3	gop.pk	1	sy	1
com.sg	15	com.py	3	org.nz	1	cern	1
pk	15	party	3	auto	1	com.nf	1
si	14	lc	3	sexy	1	gov.ie	1
gov.sa	14	gov.pt	3	herokuapp.com	1	ne	1
gov.cn	14	design	3	management	1	bc.ca	1
nu	14	nz	3	build	1	com.pg	1

so	14	ge	3	support	1	gp	1
my	14	town	3	glass	1	ki	1
sa	14	edu.eg	3	gov.tm	1	pp.ru	1
uz	13	cm	3	ac.ug	1	nr	1
tech	13	buzz	3	pet	1	pn	1
gouv.fr	13	wtf	3	guide	1	org.sa	1
sh	13	sbi	3	com.de	1	net.tr	1
wiki	13	gov.bd	3	mg.gov.br	1	cn.com	1
ma	13	run	3	fund	1	restaurant	1
ac.at	12	or.kr	3	qld.gov.au	1	madrid	1
st	12	work	3	gq	1	edu.lb	1
com.ph	12	click	3	ac.lk	1	paris	1
edu.pl	12	net.in	3	edu.ps	1	co.rw	1
com.pe	12	gov.kw	3	hosting	1	training	1
li	12	money	3	ls	1	gov.af	1
co.ke	12	co.ug	3	net.ye	1	gt	1
cx	12	police.uk	3	pink	1	basketball	1
re	12	tc	3	berlin	1	ac.rs	1
ac	12	com.mt	3	delivery	1	film	1
asia	12	co.zw	3	africa.com	1	or.id	1
media	12	sx	3	edu.ng	1	com.es	1
com.sa	11	mom	3	expert	1	xn--p1ai	1
		team	3	af	1	garden	1
						edu.mo	1