Identifying Sensitive URLs at Web-Scale

Srdjan Matic
TU Berlin

Costas Iordanou
Cyprus University of Technology

Georgios Smaragdakis
TU Berlin

Nikolaos Laoutaris
IMDEA Networks Institute

ABSTRACT

Several data protection laws include special provisions for protecting personal data relating to religion, health, sexual orientation, and other sensitive categories. Having a well-defined list of sensitive categories is sufficient for filing complaints manually, conducting investigations, and prosecuting cases in courts of law. Data protection laws, however, do not define explicitly what type of content falls under each sensitive category. Therefore, it is unclear how to implement proactive measures such as informing users, blocking trackers, and filing complaints automatically when users visit sensitive domains. To empower such use cases we turn to the Curlie.org crowdsourced taxonomy project for drawing training data to build a text classifier for sensitive URLs. We demonstrate that our classifier can identify sensitive URLs with accuracy above 88%, and even recognize specific sensitive categories with accuracy above 90%. We then use our classifier to search for sensitive URLs in a corpus of 1 Billion URLs collected by the Common Crawl project. We identify more than 155 millions sensitive URLs in more than 4 million domains. Despite their sensitive nature, more than 30% of these URLs belong to domains that fail to use HTTPS. Also, in sensitive web pages with third-party cookies, 87% of the third-parties set at least one persistent cookie.

CCS CONCEPTS

• **Security and privacy** → **Privacy protections**; • **Information systems** → *World Wide Web*; • **Networks** → Network measurement.

ACM Reference Format:
Srdjan Matic, Costas Iordanou, Georgios Smaragdakis, and Nikolaos Laoutaris. 2020. Identifying Sensitive URLs at Web-Scale. In *ACM Internet Measurement Conference (IMC '20), October 27–29, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3419394.3423653

1 INTRODUCTION

The Web is full of domains in which most people would rather not to be seen by third-party tracking services. Indeed, being tracked on a cancer discussion forum, a dating site, or a news site with non-mainstream political affinity, is at the core of some of the most fundamental anxieties that several people have about their online privacy. Many people visit such sites in incognito mode. This can provide some privacy in some cases, but it has been shown that tracking can be performed regardless as was demonstrated in recent studies [10, 38, 87].

The European General Data Protection Regulation (GDPR) [37] includes specific clauses that put restrictions on the collection and processing of sensitive personal data, defined as any data *"revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, also genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural persons sex life or sexual orientation"*. Other governments and administrations around the world, e.g., in California (California Consumer Privacy Act (CCPA) [76]), Canada [63], Israel [79], Japan [65], and Australia [62], are following similar paths [40, 44].

The above laws are setting the tone regarding the treatment of sensitive personal data, and provide a legal framework for filing complaints, conducting investigations, and even pursuing cases in court. Such measures are rather *reactive*, i.e., they take effect long after an incident has occurred. To increase further the protection of sensitive personal data, *proactive* measures should also be put in place. For example, the browser, or an add-on program, can inform the user whenever he visits URLs pointing to sensitive content. When on such sites, trackers can be blocked, and complaints can be automatically filed. Implementing such services hinges on the ability to automatically classify arbitrary URLs as sensitive and it cannot be achieved simply by installing the popular AdBlock extension or visiting the web site in incognito mode, because none of those solutions checks the actual content of web page.

At the same time, determining what is truly sensitive is easier said than done. As discussed earlier, legal documents merely provide a list of sensitive categories, but without any description, or guidance about how to judge what content falls within each one of them. This can lead to a fair amount of *ambiguity* since, for example, the word "Health" appears both on web pages about chronic diseases, sexually transmitted diseases, and cancer, but also on pages about healthy eating, sports, and organic food. For humans it is easy to disambiguate and recognize that the former are sites about sensitive content, whereas the latter, not so much. The problem becomes further exacerbated by the fact that within a web domain, different sections and individual pages may touch upon very *diverse* topics. Therefore, commercial services that assign labels to top level domains, become inadequate for detecting sensitive URLs that may appear deeper in these domains. The purpose of this paper is to demonstrate how to solve the above mentioned ambiguity problem and to develop an efficient mechanism to evaluate the extent of the sensitive content on the open Web.

IMC '20, October 27–29, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-8138-3/20/10…$15.00
https://doi.org/10.1145/3419394.3423653

Our contributions: As with all classification tasks, to train a classifier for sensitive personal data, one needs a high quality training set with *both* sensitive and non sensitive pages. Our first major contribution is the development of a semi-automated methodology for compiling such a training set by filtering the Curlie [30] crowdsourced web taxonomy project. We develop a novel and scalable technique that uses category labels and the hierarchical structure of Curlie to address the core ambiguity challenge. Our carefully selected training set comprises 156k sensitive URLs. To the best of our knowledge this is the largest dataset of its type[1].

We then consider different classification algorithms and perform elaborate feature engineering to design a series of classifiers for detecting sensitive URLs. We examine both meta-data driven classifiers that use only the URL, title, and meta description of a page, as well as classifiers that use the text of web pages. We apply our classifier on the largest publicly available snapshot of the (English speaking) Web and estimate, for the first time, the percentage of domains and URLs involving sensitive personal data. Finally, we look within the identified sensitive web pages and report our preliminary observations regarding the privacy risks of people visiting these pages.

Our findings:

• We show that classifying URLs as sensitive based on the categories and content of their corresponding top-level domain is inaccurate. This means that popular domain classifications services such as Alexa and SimilarWeb may either fail to identify sensitive URLs below non-sensitive top level domains, or mis-classify as sensitive, non-sensitive URLs below a seemingly sensitive top level domain. This should not come as a surprise, given that such services are either general purpose, or are optimized for other tasks that have nothing to do with classifying sensitive content. In essence, DNS-based blocking/domain blacklisting of sensitive content becomes ineffective at the URL level.

• On the positive side, we show that Bayesian classifiers based on word frequency can detect sensitive URLs with an accuracy of at least 88%. However, meta-data based classification, and text-based classification with do not seem to perform well. Also, word embedding techniques such as Word2Vec and Doc2Vec yield marginal benefits for our classification task.

• When it comes to detecting specific sensitive categories, such as those defined by GDPR: Health, Politics, Religion, Sexual Orientation, Ethnicity, our classifier achieves a high classification accuracy as well. For specific categories, such as Health (98%), Politics (92%), Religion (97%), our classifier achieves an accuracy that exceeds the basic classification accuracy between sensitive and non-sensitive URLs (88%).

• Applying our classifier on a Common Crawl snapshot of the English speaking Web (around 1 Billion URLs), we identify 155 million sensitive URLs in more than 4 million domains. Health, Religion, and Political Beliefs are the most popular categories with around 70 millions, 35 millions, and 32 millions URLs respectively.

• Looking among the identified sensitive URLs we reach the conclusion that sensitive URLs are handled as any other URL, without any special provision for the privacy of users. For example, we show that 30% of sensitive URLs are hosted in domains that fail to use HTTPS. Also, in sensitive web pages with third-party cookies, 87% of the third-parties sets at least one persistent cookie.

[1]For the benefit of other research efforts in the field, at the following URL we make publicly available our classifier and the categories we used to train it: https://bitbucket.org/srdjanmatic/sensitive_web/

2 EXTRACTING TRAINING DATA FROM A HUMAN-LABELED WEB TAXONOMY

The starting point for the creation of any classifier is a solid training set. This is a compelling requirement to understand the extent of the content related to sensitive personal data on the Web. In such case, the training set should be *of high quality*, *well assorted* and *large* to allow the classifier to deal with a wide range of web pages. Unfortunately, to the best of our knowledge, such a dataset is not readily available. In this section we explain how we built the training set for our classifier using hundreds of thousands of carefully selected URLs.

2.1 Limitations of Existing Commercial Taxonomy Services

Previous work relied on security solutions from vendors such as McAfee [53] and Symantec [78] to categorize URLs [8, 73, 75]. Most of these services are focused on fighting malware, and, therefore, their taxonomy includes a limited number of generic labels which categorize Effective Second Level Domains (ESLDs). Alexa [11] and SimilarWeb [74] are other extremely popular, but non security-oriented, solutions that characterize web sites at the domain level. An inherent limitation of all those approaches is that the service cannot accurately categorize subdomains that are used for different purposes than the original ESLD. In particular scenarios this might not be an issue, especially if a web site is homogeneous in terms of content, or when the objective is to characterize just the domain [69]. An example are web sites labeled as pornography where the majority of web pages actually contains pornographic material [45].

On the contrary, when the objective is to characterize individual web pages, all of the above services start having problems. This is especially true for web sites such as news portals and blogging services, that include diverse and non-homogeneous content. Limitations are further exacerbated when the categories of interest are sensitive ones. In such cases commercial services have low coverage and, even when they do, they still suffer from the ambiguity problem mentioned earlier. For example, in the Alexa top domains for Health, we find the US National Institute of health and UN World Health Organization in the top two positions. Looking at top-20 entries, we find also several fitness related web sites in the list. Being tracked while visiting such domains is probably less worrisome than when the domain relates to cancer or HIV treatment.

In addition to the coverage and ambiguity, another possible source of problems are the labels that services use. On one side those labels could be few and generic, without the ability to provide additional details (e.g., sub-categories). On the other, commercial services typically lack transparency in terms of how they assign labels to domains. Even in scenarios where such issues are not a limitation, oftentimes commercial services offer expensive APIs which are made available only to a small and targeted elite. This translates in an audience which is composed exclusively of advertisers that

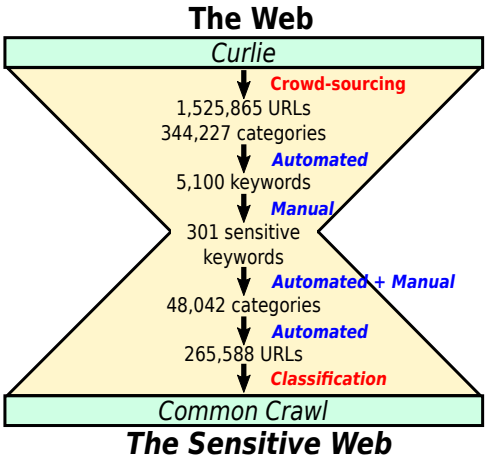

Figure 1: From Web to Sensitive Web. Mixing and matching crowd-sourcing with automated and manual filtering to create the largest ever training set for sensitive content classifiers.

want to make sure their ads are placed in appropriate contexts [2] or to proprietary solutions that work only when content is served through a specific platform [3].

2.2 The Curlie Dataset

To overcome the limitations described above, we choose to build our training set by selecting sensitive URLs from Curlie [30], the largest publicly available taxonomy of web pages. In the following sections we provide details about Curlie, its content and our methodology for distinguishing sensitive from non-sensitive web pages. Figure 1 illustrates how we blend crowd-sourcing (done by Curlie) with automated and a manual steps (done by us) on the "thin-waist" of an overall methodology that can identify the sensitive part of the Web (Section. 4). The manual step at the "thin-waist" of the overall process needs to be performed only once to identify (un-ambiguous) GDPR-sensitive categories that can then be used repeatedly to draw from the Curlie truly sensitive URLs.

What is it? Curlie is an open source project and the successor of DMOZ, a community-based effort to categorize popular web pages across the Internet [88]. Thanks to the collaboration of 92,000 editors that manually evaluate and organize web pages [29], Curlie represents one of largest human-edited directories of the Web. Editors join Curlie by applying to edit a category that corresponds to their interests, and each editor is responsible for reviewing submissions to the categories she is in charge. New editors are initially allowed to edit only a few categories, but once they have accumulated a sufficient number of edits they are allowed to edit additional areas. Community senior editors are responsible for evaluating new editors' applications in a transparent process that assures *high quality labeling of URLs* [29]. Curlie contains 3.3 millions annotated web pages, that cover 1 million different categories organized as a hierarchical ontology. At the top of the hierarchy there are the 15 top-categories visible at https://curlie.org, with the addition of a 16th, not listed, `Adult` category. Each one of the top-categories is further divided into sub-categories that provide additional granularity up to maximum depth of 14 nested layers.

Why we chose it? We chose Curlie for several reasons. First, unlike Alexa and SimilarWeb, it categorizes full URLs instead of just ESLDs. Second, the number of its categories is several orders of magnitude greater than those used by analogous commercial solutions [31, 91]. Third, the organization of the dataset in a hierarchical ontology allows us to efficiently navigate through the category tree and extract all the URLs that belong to a particular category. Finally, access to Curlie is free and not subject to any rate limitation.

Data collection. In March 2017 Curlie stopped redistributing weekly RDF[2] dumps, and, therefore, we created a crawler to download the most recent information [27, 28, 84]. We focus only on English content and thus, our crawler is seeded with the paths of the top-categories visible at https://curlie.org/en. For each seed path, the crawler performs a depth-first search to collect all the URLs included under that particular branch. It is common that a sub-category contains links to another sub-category on a completely different branch, and the crawler keeps track of all the processed categories to avoid entering loops. After completing the crawling, we collected 1,525,865 URLs that belong to 344,227 categories.

Characterizing the collected data. By inspecting how the URLs are spread across the top-categories, we notice that half of the collected URL belongs to `Regional`. This is a meta-category that acts as aggregator and groups other top-categories while providing information at the regional or country level. The remaining 15 top-categories are relatively balanced, with an average of 58,600 URLs per category. The only exceptions are `Adult` and `News` that contain less than 10,000 elements each. Such layout confirms that Curlie editors have a wide range of interests, and that the collected dataset contains enough variety for building a *well assorted* training set.

Next, we investigate the dataset coverage in terms of different web sites from which the URLs are sampled. We characterize web sites through their Fully Qualified Domain Name (FQDN), and across the entire dataset we observe 1,137,997 unique FQDNs. On average, each FQDN is represented by 1.3 URLs, but this distribution is extremely skewed and only 4% of FQDNs have *two or more* URLs. This small set of web sites contributes with 431,707 URLs, which corresponds to approximately one third of the entire dataset. This is a potential problem, because if we train the classifier with content obtained from a limited number of web sites, we run the risk of ending up with an over-fitted classifier that will not generalize well to unknown domains. To test if our dataset contains enough variety, we manually inspect the top-100 FQDNs in terms of overall number of URLs associated to them. Collectively, such web sites account for 10.4% of all the Curlie URLs, and each one of the top-32 contributors has more than 1,000 unique URLs. In Table 1 we include the top-10 FQDNs. The values in the second column show that those FQDNs are associated to thousands of categories, which in turn cover the vast majority of Curlie top-categories (third column). In the last column we point out services that allow users to participate in the creation of new content. Common examples are services where users can build their own web site (e.g., www.angelfire.com

[2]RDF or Resource Description Framework is a family of World Wide Web Consortium specifications for conceptual description or modeling of information that is implemented in web resources, e.g. URLs.

Table 1: Top-10 FQDNs contributing with the largest number of URLs. For each domain we report the total number of categories and top-categories associated to it. A ✔in the last column indicates that *multiple users* are allowed to contribute with the creation of new content.

FQDN	# URLs	# Cat.	# Top-Cat.	Multi-user
www.angelfire.com	19,918	12,970	16	✔
en.wikipedia.org	15,070	14,026	15	✔
www.newadvent.org	11,658	1,828	6	-
www.imdb.com	10,112	10,020	9	✔
www.weather.com	5,737	5,752	2	-
groups.yahoo.com	5,350	3,245	14	✔
members.tripod.com	4,731	4,044	16	✔
wunderground.com	3,929	3,918	4	-
www.facebook.com	3,435	2,833	15	✔
tools.ietf.org	3,196	99	4	✔

Table 2: Unique categories and top-categories associated to FQDNs and ESLDs with two or more URLs.

	Cat.		Top-Cats.	
# Categories	FQDNs	ESLDs	FQDNs	ESLDs
1	2,307	1,622	18,772	14,260
2	17,683	13,278	17,325	13,709
3	8,779	6,933	4,672	3,717
4	4,232	3,347	1,500	1,259
5+	10,838	9,375	1,570	1,610

and members.tripod.com) or they are encouraged to contribute by adding comments and documents on a particular topic (e.g., www.imdb.com and tools.ietf.org). A third dominant category which is not visible in Table 1, are *news websites* that account for 20% of the top-100 FQDNs. In such case, the content creators are the numerous journalists and editors. In general, those three types of service are extremely popular and 84% of the top-100 FQDNs belongs to one of them. The remaining 16 FQDNs are specialized services offering information on aviation, plants and hotels. After isolating the URLs associated with these 16 FQDNs, we observe that they account only for 1.4% of the dataset, and thus their impact on the training set can only be marginal.

As final assessment we study the differences among per-URL and per-domain categorization. Our goal is to understand the possible benefits of having categories assigned to individual URLs instead of using a the same category for all the elements under an ESLD. To this end, for all the FQDNs associated to *two or more* URLs, we extract the corresponding ESLDs as well as the overall number of categories assigned to those domains. The results of this process are shown in Table 2. When we use full category names only 5% of all the URLs under a particular FQDNs or ESLDs belong to the same category. This is somehow expected since Curlie has more than 340,000 extremely fine-grained categories. These values change significantly if we adopt a more coarse-grained grouping, using the top-categories; in this case 42% of the URLs under a particular domain belong to the same category. Despite having only 16 top-categories, and even if they include extremely generic types such as `Society` or `Regional`, still 18% of the domains are flagged with at least three different top-category names. Those results suggest that any commercial solution that uses a unique category for all the URLs under the same ESLD, in 58% of the cases would erroneously categorize *at least one* of the URLs. Our analysis on Curlie demonstrates that the collected dataset (i) contains enough variety for building a well-assorted training set, and (ii) offers significant advantages compared to commercial solutions. In the next section we explain how we leverage this dataset to create the training set for our classifier.

2.3 Building the Classifier Training Set

Using the Article 9 of GDPR we define *five sensitive categories* that include Ethnicity, Health, Political Beliefs, Religion and Sexual Orientation. According to GDPR, the collection and processing of information about any of these categories should be subjected to special rules [36]. Our goal is to create a classifier that can identify web pages that belong to those five sensitive categories. To this end, we first identify the Curlie categories that are related to the 5 sensitive categories under GDPR, and then we collect the resulting URLs from Curlie. Finally, we download the content associated to those URLs and use it as training set for our classifier.

Identifying sensitive categories. Curlie contains hundreds of thousands of categories and, thus, we cannot simply inspect them *manually* to determine which ones meet the requirements for being considered *sensitive*. We cannot leverage the organization of categories into a hierarchy, because we do not know the maximum depth at which to stop the exploration without missing elements contained in deeper branches (e.g., the category `Regional/United_States/Illinois/Localities/C/Chicago` might seem not relevant to health without knowing that it contains a sub-branch called `Addictions`). Finally, as we do not have a list of descriptive keywords associated to each category, we cannot either select URLs by looking in the web pages for those keywords. It is extremely challenging to craft such a list because the keywords we might choose might not be representative for the Curlie dataset. For example, a lookup of the "LGBT" across the Curlie dataset, generates a set of 240 URLs, while searching for the keyword "gay" selects 3,873 URLs. By using our own list we incur in the risk of including many ambiguous keywords (e.g., "virus") that characterize both sensitive and non-sensitive content, or others that are too specific (e.g., "HIV"). In both cases the consequence of the inability to include enough elements for particular categories, would result in a significant loss in performance or even the impossibility to use the classifier on other datasets.

We develop a technique that extracts *structured knowledge* from the Curlie dataset, see Figure 1, and we use it to generate our training set. Our approach leverages the names that Curlie editors choose for their categories to detect relevance with sensitive categories. In detail, we first create a list of all the keywords included in the names of the Curlie categories. Next, by selecting all the categories that contain a particular keyword, we associate a keyword to the list of URLs under those categories. For example, let's assume that the dataset contains only three categories `Health`, `Health/Addictions/Food` and `Health/Animals/Food` with respectively 3, 100 and 20 URLs. In such case, the list with the counting of the URLs associated to each keyword would be: (`Health`, 123), (`Addictions`, 100), (`Food`, 120), (`Animals`, 20).

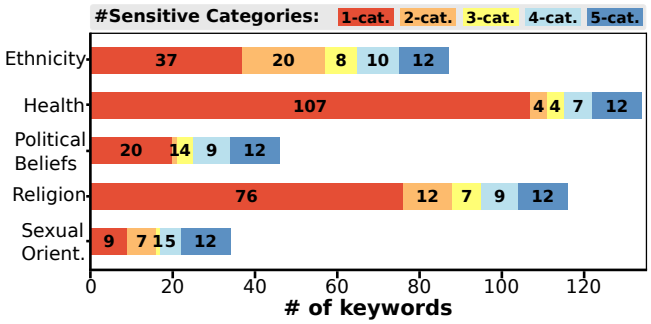

Figure 2: Sensitive categories and corresponding set of keywords. For each set of keywords, we report how many other sensitive categories might be associated to this same set.

After applying this process on the entire Curlie dataset, we obtain a set of 110,475 unique keywords. Next, we manually inspect those keywords to identify those that could be *could be potentially associated* to sensitive categories. For this process we restrict the focus on a subset of 5,1000 most representative keywords, which are associated to at least 100 URLs. Moreover, we apply a greedy approach and we include as many generic keywords as possible (e.g., "Health") while discarding only those that are unlikely to be associated with any sensitive content (e.g., "Animals"). At the end of this analysis, we generate a set of 301 carefully selected keywords which we annotate with all of the sensitive categories that could be connected to them. Some keywords are extremely specific (e.g., "Judaism"), while others could be linked to multiple sensitive categories (e.g., "Communities"). In the final step, we *manually* inspect 48,042 the Curlie categories where the 301 keywords appear, and we verify if they are indeed sensitive. When a Curlie category is confirmed to be sensitive, all the URLs contained under this category are added to the corresponding GDPR sensitive category. This is a slow and time-consuming activity, but is necessary to ensure that all the URLs are included in the correct category. Furthermore, this manual step needs to be done only once, and we can then re-draw URLs from Curlie with high confidence that they will indeed be sensitive. For some elements we were able to assess the sensitivity of the URLs by leveraging only the keyword that appears in the category name (e.g., all the URLs under the 553 categories that embed the "Local_Churches" keyword). In other cases, we inspected the structure of the Curlie categories together with the location of the keyword. For example, of the 13,299 categories that include the "Health" keyword, 4,927 were under the "Regional" top-category and "Health" always appeared as final sub-category (e.g., `Regional/Asia/India/Punjab/Health`). After checking a few dozen samples, in all those cases the URLs were pointing to local services or clinics and we label all the URLs under those 4,927 categories as health-related. To facilitate the analysis, we use several tricks including: sorting alphabetically the categories, leveraging the information from sub-categories, checking URLs strings and web page content. As side effect of the manual validation, all the URLs in categories that are not sensitive are included into a *sixth Non-sensitive* category.

Table 3: Content retrieved from the URLs of our training set, grouped into the GDPR sensitive categories.

GDPR Cat.	#URLs	GDPR Cat.	#URLs
Ethnicity	9,547	Health	59,025
Pol. Beliefs	15,668	Religion	68,625
Sex. Orientation	3,924	*Non-sensitive*	*64,923*

Figure 2 shows how representative the 301 keywords are for each sensitive category. With respect to the specificity of the keywords, we notice that in general 249 of the keywords (the red boxes in the figure) are unambiguous and they uniquely identify only one category. This is the case of "Local_Churches" or "Marxism", which immediately recall to the sensitive categories "Religion" and "Political Beliefs". Generic terms such as "Clubs_and_Associations" or "Organizations", that can be related to several sensitive categories, appear to be rare and account only for 4% of all the keywords. Not all of the sensitive categories have an equal number of keywords and some categories are less represented. Moreover, categories such as Ethnicity, Political Beliefs and Sexual Orientation also contain higher percentage of generic keywords which can be associated to multiple sensitive categories (the non-red boxes in Figure 2). A direct consequence is that these sensitive categories are less likely to generate many candidates for the corresponding Curlie categories. At the end of the validation process, we are left with 265,558 URLs, each one tagged with at least one GDPR-sensitive category, drawn from 48,042 Curlie categories.

2.4 Final Labeled Dataset

After successfully mapping the GDPR sensitive categories on the Curlie dataset, we use the labeled URLs to download the content to train our classifier. As first step, we filter out all the URLs that received more than one sensitive category as label. Since those multi-category URLs will not be used to create the training set we can avoid download the corresponding content. Next, we connect to each URL from four different locations, two in Europe and two in US, to maximize the likelihood of obtaining the content. We also apply a mechanism to detect the presence of error pages, to avoid training the classifier with spurious content. To this end, from each web site associated to a URL, we download also a randomly generated resource. The intuition behind this is that a request to a non-existing resource will likely return an error page. We build a list of hashes for the error pages, and we filter any content that results being an error page. After this step, we generate a dataset with 221,712 web pages that we use to train our classifier. The dataset contains the five sensitive categories defined by GDPR, as well as a sixth non-sensitive category. In this additional category we include all web pages that our manual validation confirm that do not belong to *anyone of the five GDPR-sensitive categories*. Table 3 shows the GDPR categories and the number of URLs associated to each category; each URL belongs to only one GDPR category. Health and Religion are the sensitive categories with the highest number of URLs.

3 BUILDING A CLASSIFIER FOR SENSITIVE WEB PAGES

In this section we describe how we build an accurate classifier for identifying sensitive web pages. Our objective is not to propose a new text classification method, but rather combine existing work in the area [12, 89] in the best possible manner for our goal.

3.1 Designing the Classifier

To develop the classifier we test different options for the algorithm, the data preprocessing step, and the feature selection.

Classification algorithms. There is a wide range of popular algorithms that are suitable for classifying web pages. Examples are K-Nearest-Neighbors [19, 41, 49], Naïve Bayes [32, 33, 48], Support Vector Machines [21, 22, 77, 90], Decision Trees [35, 80, 85], Neural Networks [42, 56] and different variations [59], maximum entropy [23, 49]. Given the scope of this work, we choose the Naïve Bayes classification algorithm for the reasons explained in Section 3.2.

Input data. The input dataset consists of the Curlie web pages identified using the approach outlined in Section 2.4. As first step, we exclude from the HTML code any non-visible element (i.e., JavaScript, CSS, etc.) except for the HTML <META> tag. Note that the <META> tags include a short list of keywords describing the page content. We extract all the text from the visible content, we call this input source "web page content" (C). Similarly, we refer to the content obtained from the <META> tag as "meta-data" (M).

Data preprocessing. We apply standard text preprocessing steps on both the web page content and the meta-data. Such steps include (1) the transformation of all letters in lower case and (2) the removal of stop words. In addition (3) we also impose a minimum word length to three letters, numbers, or any combination of the two; and (4) we remove content with less than 1,000 characters.

Feature engineering. With respect to feature engineering, we test a wide variety of algorithms. Due to space limitations, in Section 3.3 we present only the results for the three algorithms with the best performances.

3.2 Selecting and Training the Classifier

Our classifier has to be applied at Web scale while efficiently detecting web pages that belong the five sensitive categories as defined by GDPR. To this end, we choose a multinomial Naïve Bayes algorithm, because it allows us to train a single supervised classifier that can predict multiple classes. Our choice is based on several reasons. First, it has a simple and easy training and classification stages [77]. Second, it is a fast learning algorithm that can handle large numbers of features and classes [21, 55]. Third, the algorithm was already tested with good results on older versions of Curlie for a multiclass web page classification purposes [64]. Fourth, the algorithm showed comparable, and, in some cases, better performance than other classifiers [9, 81]. Finally, several off-the-shelf implementations are publicly available, making it easier for other researchers to reproduce and validate our results.

We train the classifier using the training set described in Section 2.4. Such set contains 221,712 web pages, with the corresponding GDPR category as label. From each web page we extract both the human-readable text and the meta-data information. Next, we filter the content by applying the preprocessing steps described in Section 3.1. This procedure generates a final set of 218,696 URLs with content. Finally we use Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec [60] & Doc2Vec [50] to extract the features.

Bag-of-Words (BoW) [46] is a popular Information Retrieval (IR) technique that represents texts as a multiset of words. When such technique is applied, the classifier disregards grammar rules, but keeps track only of the word multiplicity (i.e., the number of occurrences of a word within a single document or a corpus).

Term Frequency-Inverse Document Frequency (TF-IDF) [72] is a popular IR numerical statistic which captures the importance of a word within a document. The TF-IDF value increases proportionally to the occurrences of the word within a document, and inversely proportionally to its frequency across other documents.

Word2Vec & Doc2Vec [50, 60] are word embedding techniques that take into consideration both the semantic meaning and the order of words with in a given text. Word2Vec is generally applied at the paragraph level, while Doc2Vec uses information obtained from the entire document. In our case the documents are web pages contents, and we leverage word embedding to extract the keywords that are used to train the classifier. It can be used as an intermediary stage in our case to extract key words and use them as features during the training phase of the classification algorithm.

For all three feature extractions algorithms that we test above, we keep all hyperparameters to their default value as defined by each corresponding Python library, that includes, BoW [5] and TF-IDF [6] from the sklearn (ver. 0.21.3 [7]), and Doc2Vec from gensim (Ver. 3.8.1 [4]) library. In the next section we discuss the results of each algorithm and we explain the additional optimizations we apply.

3.3 Classification Accuracy and Optimizations

Classification accuracy is defined as the percentage score, from 0% (lowest) to 100% (the highest), that a classifier can accurately assign items (pages) to their correct category. The accuracy is influenced both by the choice of the input data and by the algorithm used to extract the features. To identify the combination of input data and the algorithm that leads to the highest accuracy, we start by restricting the set of feature only to 5k elements. We then apply each algorithm, using as input data the web page content, the meta-data, but also their combination. During this process, we reserve 70% of the input for the training phase and the remaining 30% for testing. To avoid any bias, we also repeat our experiments using 10-fold validation obtaining consistent results.

Feature selection. Table 4 shows the accuracy for different combinations of the input data and the algorithms using 5k features. When we use only web page content (C), the average accuracy of the three algorithms is around 81%. TF-IDF and Doc2Vec obtain nearly identical results, while the accuracy for BoW is slightly lower, around 78.5%. When the input is only the meta-data (M), the accuracy drops for two out of the three algorithms. In the case of Doc2Vec, the accuracy is just above 56% if we use meta-data as input. This result is a direct consequence of how the Doc2Vec algorithm works, because it leverages full sentences to capture the

Table 4: Classification accuracy using 5k features and all the possible combinations of input data and algorithms to extract the features.

	Feature Engineering		
Feature Source	BoW	TF-IDF	Doc2Vec
Content (C)	78.48%	82.17%	82.34%
Meta-data (M)	78.55%	79.62%	56.51%
C + M	79.90%	83.33%	82.77%

Table 5: The top-10 features that the classifier uses to determine the category of each web page.

Rank	Health	Ethnicity	Religion	Sexual Orientation	Political Beliefs	Non-sensitive
1	Health	Genealogy	Church	Sex	Election	Club
2	Care	Family	Catholic	Gay	Party	Association
3	Dental	County	God	Porn	State	Home
4	Medical	Tree	Worship	Material	District	Members
5	Services	History	Bible	Adult	Democratic	Events
6	Treatment	Records	Sunday	Fetish	Democrats	News
7	Patients	Indian	Christ	Escorts	Senate	Membership
8	Surgery	Genealogical	Christian	Sexual	Political	Contact
9	Therapy	American	Jesus	Lesbian	Government	Read
10	Dentistry	Native	Ministry	Nude	Republican	Society

semantic relevance between adjacent words. Most of the times the meta-data tag contains a list of keywords, in random order, that describe the page content. Moreover, the same keyword can appear in multiple uncorrelated lists across different web pages. The last row of the table depicts the results for the combination of the two different input data (C+M). In this scenario, we achieve the highest accuracy when we extract features using the TF-IDF algorithm. When we compare those results with the classifier that uses only web page content, we observe that across all the feature extraction algorithms we have an average increment of 1% in accuracy. We chose TF-IDF on input data C+M as baseline for our evaluation and refer to it as *baseline classifier*.

Feature sets. To understand the classifier robustness, for each class we sort the 5k feature vector and we check what are the features that received the highest weights. In Table 5, we list the top-10 features across the six different categories. We observe that for each one of the five sensitive categories the top-10 most important features are well suited to characterize the category. In the case of Ethnicity we notice some bias toward *Indian Americans*; and a similar behavior appears also with terms related to *Christianity* in the Religion category. We attribute such bias to the fact that from the Curlie dataset we extract only English content, that are associated to the Western culture and more specifically United States. Similarly, in the Non-sensitive category we see mainly terms linked to organizations and social activities. Also in this case, the result could be explained if we consider the Curlie dataset from the perspective of the top-level categories, as we did in Section 2. We attribute this bias to the fact that 80% of the web pages originate from the largest Curlie top-level categories, namely `Regional`, `Art`, `Society` and `Business`. Those four top-categories can be easily connected with the 10 most representative terms that the classifier associates to the Non-sensitive category.

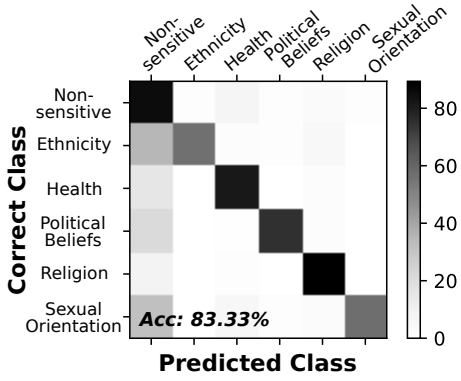

Figure 3: Confusion matrix of the baseline classifier.

Accuracy of individual categories. Figure 3 presents the confusion matrix that visually summarizes the classification accuracy for each sensitive category. The rows of the confusion matrix contain the instances of a specific class, and columns represent the *prediction* of the classifier. Shades on cells indicates the percentage of elements that are predicted belonging to a particular class: white cells indicate lower percentage values, darker cells higher ones. In an ideal confusion matrix, all the cells are white except for the elements on the main diagonal, which are all black. In such case, the classifier always predicts the correct label for all of the input elements. In our matrix we observe darker cells only for half of the categories (i.e., Non-sensitive, Political Beliefs and Religion). In the remaining three cases, the lighter coloration suggests that the instances are spread among the correct and at least another class, typically the Non-sensitive. This is particularly evident in the first column that contains the highest concentration of gray cells. Such trend indicates that *all the five sensitive* categories, with different degrees, have some elements that get mis-labeled as Non-sensitive. We also observe a small percentage, around 11%, of Non-sensitive web pages that occasionally get labeled either as Health or Religion. The fact that the majority of the mis-classifications are localized on the first column, can be more, or less damaging, depending on the particular use case. For example, since our goal is to build a framework that can detect sensitive web pages across the Web, we can use this kind of classifier to derive a conservative estimation on the number of URLs and domains hosting sensitive content. In a similar use case, the law enforcement agencies might leverage this classifier, combined with third-party detection tools and methodologies, (see [43]) to check GDPR compliance for a large number of web pages. In such case the penalty is much higher if instead of just missing some elements, the analyst has to manually go through tens of thousands of web pages with legitimate content erroneously marked as sensitive. Our results also show that the mis-classification of non-sensitive web pages to sensitive categories is low, which is a desired outcome as we do not want to penalize non-sensitive web pages.

3.4 Balancing the Classifier

Up to this point all the presented results have been produced by applying the baseline classifier and using the dataset discussed

Table 6: The number of Curlie web pages used per classifier. "Initial": baseline classifier , "Final": balanced classifier , "Added": additional URLs included after applying baseline classifier on the URLs in *Unknown* set.

	URLs		
GDPR Cat.	Initial	Added	Final
Ethnicity	9,399	+ 4,710	14,109
Health	58,533	+ 16,231	74,764
Pol. Beliefs	15,543	+ 21,646	37,189
Religion	67,593	+ 23,541	91,134
Sex. Orientation	3,651	+ 248	3,899
Non-sensitive	63,977	+ 157,118	221,095
Unknown	1,060,077	- 223,494	836,583

in Section 3.2. In this section, we explain how to improve both accuracy and coverage by adding more variety to the training set.

Table 6 offers a detailed overview of the contributions of each category, after applying the preprocessing steps discussed in Section 3.1. The vast majority of the web pages belong to the *Unknown* category, that includes URLs from Curlie categories that we did not manually validate. For this reason we cannot use them to test our classifier. Since our manual validation technique was specifically designed for identifying sensitive web pages, we were able to cover only 17% of all the Curlie URLs. At the same time, since we want to deploy the classifier on the Web, we would like to train it on a dataset that is *as large as possible*. In addition, our original training set contains unequal proportions of sensitive and non-sensitive web pages. This stems from the fact that the initial dataset is build leveraging the 301 "potentially sensitive keywords", and then following the technique outlined in Section 2.3. Because of this, the sensitive elements outnumber the non-sensitive ones by a factor of 2.4. Having the majority of web pages be sensitive, can lead to over-fitting and poor performance over different sets of web pages. This is particularly true for a Naïve Bayes classifier, which is known to have performance problems with datasets involving unbalanced classes [47]. To overcome these problems, we train a second classifier using the same ratio of sensitive and non-sensitive elements. To create such a balanced training set, we first run our baseline classifier on the web pages that we did not validate manually (those with the Unknown label). From the 1,060,077 Unknown URLs we extract elements that we use to augment each individual category, and at the same time, balance the ratio between sensitive and non-sensitive ones. A detailed overview of the URLs that are included in each category is available in Table 6. In Appendix A we provide additional details on how we extract sensitive URLs from the Unknown elements and how we validate their correctness.

The final "balanced" dataset contains 442,190 web pages, equally split among sensitive and non-sensitive ones. Using this dataset, we train a second classifier, that we will henceforth call *balanced classifier*. Figure 4 presents the confusion matrix for this balanced classifier and Table 7 reports the percentage values of each cell of the confusion matrix alongside the Precision, Recall and F1 scores

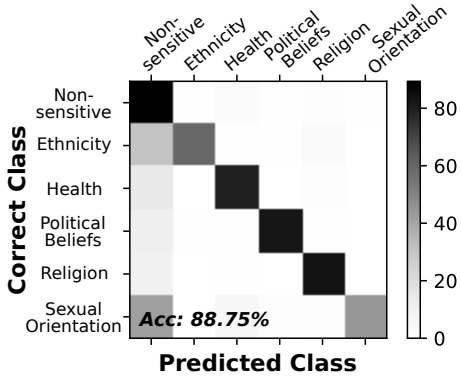

Figure 4: Confusion matrix of the balanced classifier.

for each individual class (Last three columns from left to right). We observe several benefits compared to the same matrix build for the baseline classifier (Figure 3). First, the overall accuracy increases by 5.2%. Second, only 6.2% of all the non-sensitive web pages are occasionally labeled as sensitive. Third, the number of web pages related to Political Beliefs that are mislabeled as non-sensitive drops by half (from 21.9% to 11.7%). In the Sexual Orientation category, the amount of web pages now considered non-sensitive increases by 11.6%, which corresponds to a total of 173 new elements that pass undetected. Given that Political Beliefs set is ten times bigger than Sexual Orientation, the benefits for the former outweigh the penalties for the latter. For the three remaining categories, the percentages remain consistent with those obtained using the baseline classifier. It is also worth noticing that the mis-classification of non-sensitive web pages to sensitive categories decreases by 8.4% with the new classifier. Nevertheless, for the Sexual Orientation and Ethnicity classes, the F1 score remains low at 0.55 and 0.73, respectively (Table 7 - last column)

3.5 Sensitivity to the Number of Features

In Figure 5 we report several performance metrics obtained by gradually increasing the size of the feature vector of the balanced classifier. We include the overall accuracy, the precision and recall for each category, as well as their combination (F1 score). With respect to the accuracy, we observe a marginal increase up to 1% when we configure the classifier to use larger feature vectors. Such increase is monotonic when using vectors with less than 10k features, and stabilizes after the vector size has reached 30k elements. For vectors with more than 30k elements the overall accuracy starts gradually dropping following a trend that is inverse to the number of features that are used. For the majority of the categories, increasing the size of the feature vector has positive effects both on the precision and the recall. The main category in which we observe a significant drop for both statistics is Sexual Orientation. This category contains the smallest number of samples, and when the feature vector becomes very large (i.e., more than 20k elements) the classifier starts over-fitting causing negative impact on the recall. A similar behavior is observed for Ethnicity, where increasing the number of feature improves the precision, but it affects negatively the recall. Both those groups contain a significantly smaller

Table 7: Quantitative results of the balanced classifier depicted in Figure 4.

	Non-sensitive	Ethnicity	Health	Political Beliefs	Religion	Sexual Orientation	Precision	Recall	F1-Score
Non-sensitive	93.8%	0.4%	3.3%	0.8%	1.3%	0.4%	0.87	0.94	0.90
Ethnicity	32.5%	62.5%	0.5%	1.1%	3.3%	0.001%	0.86	0.63	0.73
Health	15.3%	0.001%	83.2%	0.3%	1.2%	0.001%	0.88	0.83	0.85
Political Beliefs	11.7%	0.5%	0.5%	86.5%	0.8%	0.001%	0.93	0.87	0.90
Religion	10.7%	0.2%	1.0%	0.3%	87.8%	0.001%	0.94	0.88	0.91
Sexual Orientation	44.1%	0.0%	6.0%	1.7%	2.0%	46.3%	0.68	0.46	0.55

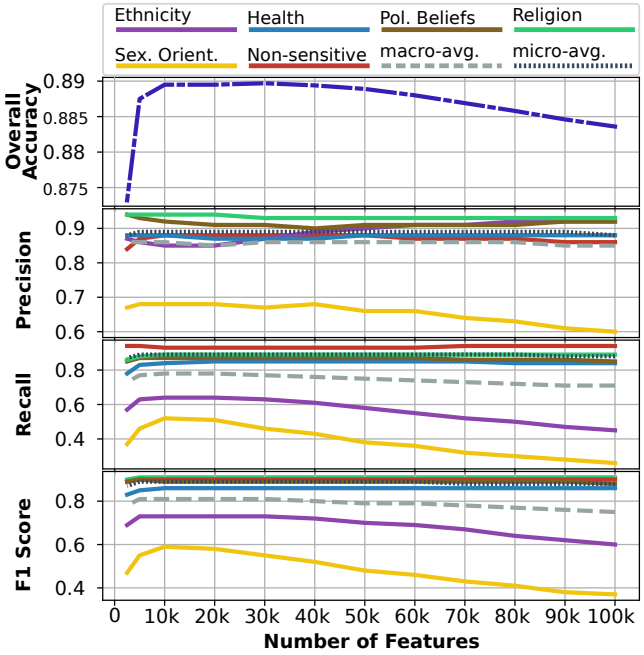

Figure 5: Accuracy, precision, recall and F1 scores Vs. number of features for the balanced classifier.

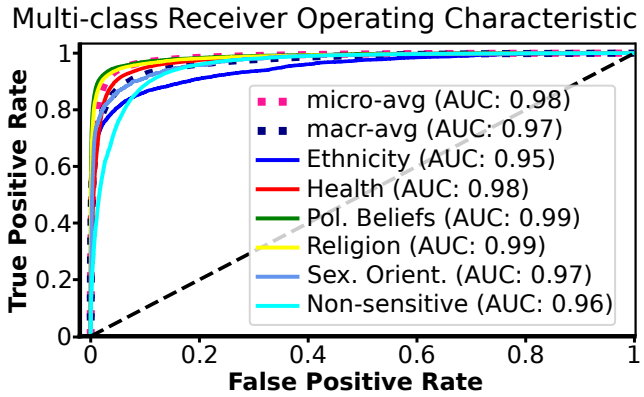

Figure 6: The micro and macro-average AUC using the balanced classifier and the corresponding breakdown per category.

number of samples compared to other categories. This unbalance is reflected in the scores of the macro- and micro-average. Micro-average obtains much better results thanks to the fact that larger categories such as Non-sensitive, Health and Religion have much higher precision compared to those with fewer samples.

To select the most appropriate length of the feature vector, we turn to the F1 scores. Using larger vectors, up to 30k features, generally benefits the overall scores (both micro- and macro-weights increase). Unfortunately this behavior is limited only to categories with a higher number of samples. The recall for Ethnicity drops below 60% for vectors with more than 20k features, and in the case of Sexual Orientation the peak value is observed when vectors contain less than 20k elements. In an attempt to find the ideal trade-off among the six categories, we decide to use a feature vector with 20k elements. We choose such value because it is very close to the peak value for four of the largest categories. In addition, such choice allow us to maximize the overall accuracy, while not sacrificing the recall in categories with fewer samples.

In Figure 6 we plot the Area Under the Curve (AUC) for the balanced classifier as well as the AUC values for each individual sensitive category. The AUC depicts the performance of the classification model at all classification thresholds (cut points), as opposed to the overall accuracy that is based on a specific threshold, and can provide an aggregate measure of performance across all possible classification thresholds. We observe that the micro- and macro-average AUC values are 0.98 and 0.97 respectively. With respect to the individual categories, we observe that the lower AUC value at 0.95 belongs to the category Ethnicity followed by Non-sensitive at 0.96. Sexual Orientation is at 0.97 and Health at 0.98. Finally, Political Belief and Religion show equal AUC value at 0.99. Our analysis shows that the balanced classifier achieves very high accuracy. For the rest of the paper we will use this classifier to analyze corpuses of the Web to identify sensitive web pages.

4 SEARCHING FOR SENSITIVE WEB PAGES IN THE WILD

We leverage our classifier to investigate the popularity of sensitive content across the open Web, and we analyze the privacy and security practices associated to this type of content. To perform our study we use data obtained from a service with more than a decade of experience in crawling the Web. In the following sections we provide an overview of the dataset, we share our experience in classifying Billions of web pages, we evaluate the extent of sensitive web pages and we report potential privacy issues for the users accessing this type of content. Our analysis identifies around 155

Table 8: English-only web pages in the Common Crawl October 2019 snapshot after excluding duplicates, error pages and content with less than 1,000 characters.

	URLs	FQDNs	ESLDs
HTTP	276,876,278	6,561,287	5,286,123
HTTPS	709,263,254	8,005,488	6,294,040
HTTP+HTTPS	-	540,230	586,622
Homepages	9,148,978	9,148,978	7,827,525
Total	986,139,532	15,107,005	12,166,785

million sensitive URLs in more than 4 million domains, with Health, Religion, and Political Beliefs to be the top-3 sensitive categories.

4.1 The Common Crawl Dataset

Common Crawl is a nonprofit organization that maintains an open repository of Web crawl data [25]. The project was launched in 2007 and since then it periodically releases all the collected information in the form of monthly snapshots. All the data is made publicly available and a single snapshot contains more than 3 Billions of web pages [26].

Common Crawl October 2019 corpus. Each corpus (or snapshot) contains a list of URLs and corresponding web pages, with the addition of metadata about crawling. This information is packaged in the Web ARChive (WARC) format, combining the raw HTML content together with the HTTP headers fetched from servers. As alternative, users can choose the WET archives that contain only plain text extracted from the raw crawls, once HTML tags are removed. Since our classifier works with web page content, we speed up the analysis by downloading the WET archives for the October 2019 snapshot. Before starting to classify content, we leverage the Common Crawl language annotations [24] to identify documents written in English language. We also apply a preprocessing similar to the one used for the Curlie dataset, to remove extremely short documents and error pages.

In Table 8 we present an overview of the October 2019 snapshot, after applying our preprocessing steps. The dataset contains almost 1 Billion web pages in English, collected from 15,107,005 different web sites (column FQDN). For 60.5% of the visited web sites the crawler successfully downloaded the homepage. Almost 72% of the pages were downloaded through HTTPS and the vast majority of web sites was accessed using a single protocol; only a small fraction of 3.4% of FQDNs was accessed both with HTTP and HTTPS. From 60% of the FQDNs the crawler collected at most ten URLs, and in 30% of the cases a web site is represented only through a single URL. However, the distribution has a long tail and 115,084 FQDNs in the snapshot contribute with more than 1,000 URLs each.

To check the popularity of the domains that are included in the snapshot, we use the list from the Tranco project [66]. Such list aggregates the ranks from four widely used services which provide daily updated lists of popular domains. We use the Tranco list generated on the 31 October 2019[3] which covers the same period when the Common Crawl snapshot was created. By comparing the two sets of domains, we notice that 475,637 of the FQDNs that appear in the snapshot are also included in the Tranco list of the 1 million most popular domains. This confirms that the Common Crawl project collects information from well know and popular web sites, together with other services which have less visibility.

[3]https://tranco-list.eu/list/5XQN/1000000

Table 9: Classifier results on the October 2019 snapshot. We report the percentage of URLs in each category and the FQDNs associated to those URLs. In the last column we report the FQDNs where all the URLs belong to that category.

Category	% URLs	**# FQDNs**	% URLs	**# Ded. FQDNs**
Ethnicity	0.52	**112,300**	0.03	**12,935**
Health	7.1	**2,782,416**	1.78	**922,242**
Pol. Beliefs	3.28	**686,733**	0.17	**95,848**
Religion	3.59	**1,228,243**	0.75	**329,575**
Sexual Or.	1.29	**214,011**	0.55	**86,345**
Non-sensitive	84.22	**13,605,876**	38.11	**10,853,118**
Mixed			58.28	**2,752,758**

Classifying the Common Crawl corpus. The 3 Billion web pages of the October 2019 snapshot are partitioned into 56k zipped archives, totaling 10 Terabytes of disk space. To classify the snapshot we used a mid-level server with 30 cores and 192 GB or memory. To make sure that I/O operations on the hard drive don not become a bottleneck, we developed a framework with a dispatcher module that coordinates a pool of workers that dynamically process the files. The dispatcher iterates over all the archives, loads them in memory in their uncompressed format, and assigns each pointer in memory to a worker. The worker extracts the file, removes error pages, non-English documents and any content with less than 1,000 characters. In the final step, all the contents that are not filtered out receive a probability score for each one of the six categories presented in Section 3. As soon as the worker finishes processing a file, it contacts the dispatcher which replies with the next file to be processed. To classify the entire October 2019 snapshot our framework took ≈86 hours. After the classification, we manually assessed the classifier accuracy by sampling around one hundred URLs for each category and verified that the average accuracy of the classifier was above 90%.

4.2 Analysis of the October 2019 Corpus

Table 9 shows the breakdown of classified pages into different categories. For each category, we report all the URLs with a specific label and the FQDNs associated to those URLs. In the third column we include only the subset of URLs that belong to *dedicated FQDNs*. We use this term to refer to FQDNs in which *all of their web pages belong to the same category*. Similarly, in the *Mixed* category we include only FQDNs that at the same time served both *sensitive and non-sensitive* content. Each of those FQDNs contains at least two URLs: one sensitive and another non-sensitive. Across all the categories, the URLs are reported as a percentage of the 986,139,532 web pages contained in the snapshot. Unsurprisingly, the vast majority of elements in the snapshot is non-sensitive, whereas sensitive pages account only for 15.78% of the URLs. Such value is very close to the 17.29% of sensitive content that our balanced classifier detected in the Curlie dataset. Within the sensitive categories the

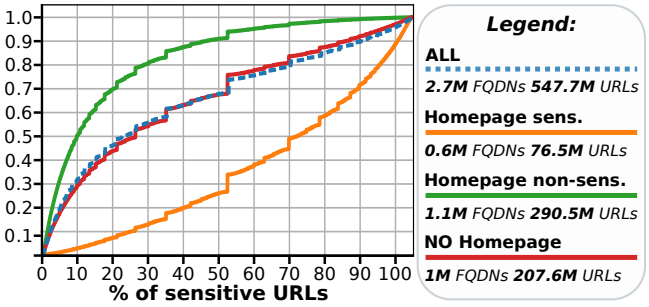

Figure 7: Cumulative Distribution Function with the percentages of web pages with sensitive content across the FQDNs with mixed content.

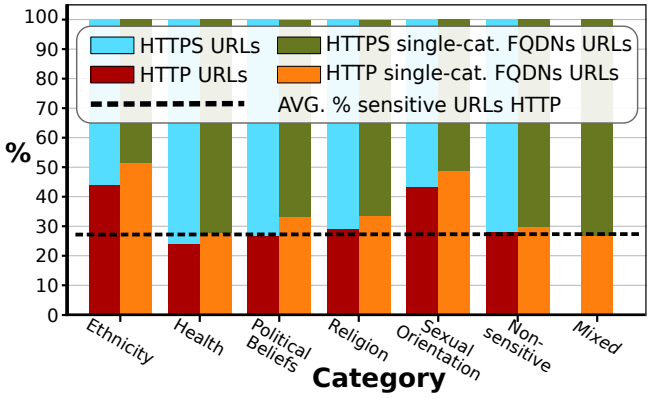

Figure 8: Protocol adoption for the URLs associated to the eight categories in Table 9.

highest number of labels originates for Health (70 millions), followed by Religion (35 millions) and Political Beliefs (32 millions). By comparing the percentages of URLs in the second and third column, we notice that Sexual Orientation and Health are the categories with the highest concentration of URLs hosted on dedicated FQDNs. This suggests that pages related to these categories are much more likely to identify websites where the majority of the web pages deal with similar topics. The exact opposite happens for Ethnicity and Political Beliefs, in which URLs are spread across a wide range of FQDNs with different content. Overall, we found sensitive content among 28% of the 15 millions domains included in the snapshot and at least one sensitive URL in 97% of FQDNs included in the Tranco list of popular domains. The 2,75 million FQDNs with mixed sensitive and non-sensitive elements are responsible for 58.28% of the snapshot content, which confirms that black-listing (white-listing) web pages based solely on FQDNs is bound to produce lots of false negatives (positives).

Mixed category. To determine how pervasive sensitive content is across this category we split the 2.75 millions FQDNs into web sites for which the homepage was available in the snapshot, and those for which it was not. In case the homepage was included, we further group FQDNs into those with sensitive and non-sensitive homepages. Figure 7 depicts the percentage of sensitive elements across the three groups of FQDNs. The largest group is one with FQDNs with a non-sensitive homepage, and half of those FQDNS contain at most 10% of sensitive URLs. Only a small fraction of FQDNs, around 9%, have more than half of their content labeled as sensitive. We observe the exact opposite trend in FQDNs that have a sensitive homepage where half of the FQDNs have more than 70% of their URLs labeled as sensitive. A possible explanation is that sensitive web sites usually refer to a smaller set of topics than generic, non-sensitive, ones. For example, in presence of a homepage promoting religion or discussing a particular disease it is extremely likely that other web pages on the web site will be addressing the same subject. For FQDNs where the homepage was not available in the snapshot, the percentage of sensitive content is an average of the other two cases. Overall, we notice that around half of the FQDNs with mixed content at most 20% of their URLs labeled as potentially sensitive, and that in presence of a homepage marked as sensitive such percentage goes up to 70%. Those results suggest that if a sensitive URLs has been identified, it is likely that the web site will be hosting additional sensitive web pages.

Categories and protocols. We investigate possible correlations among sensitive categories and the choice of the protocol that is used to serve the content. To this end we compare dedicated and non-dedicated FQDNs serving URLs that belong to each sensitive category. The results of this analysis are presented in Figure 8. Across all the categories the relative percentage of URLs offered through HTTP is always higher on dedicated FQDNs. An explanation could be that in presence of dedicated FQDNs we analyze fewer domains and for this reason we are not able to observe the global picture. Another possible source of the bias could be related to the hyperlinks that the crawler followed and the method that was used to fetch the URLs. With those caveats, we observe that all sensitive categories exhibit a similar behavior which differs from the non-sensitive and mixed FQDNs. All sensitive categories excluding Health, seem to choose HTTP as preferred protocol and in the case of Ethnicity and Sexual Orientation nearly half of the content is offered over HTTP. Even if those results are enough to draw a strong correlation among the category and the protocol, they suggest that owners of dedicated web sites do not seem to put special efforts in protecting access to potentially sensitive content, and that those URLs are handled like any other URL.

4.3 Preliminary Observations on the State of the Sensitive Web

We conclude our analysis with a study on cookie usage across the categories. Similarly as we did for protocols, our goal is to understand the way different categories handle cookies, and to check if some categories adopt stricter policies. To this end we use the categories from Table 9 and from each dedicated FQDN we sample up to 5 URLs. In total we select 700,000 URLs and we visit them with a framework that leverages a fully-fledged browser [57]. For each page we wait 60 seconds, and we do not perform any action, mimicking a new user that accesses the web page for the first time without giving consent for the installation of cookies. The experiments were performed during the first months of 2020 from two different locations, one in Germany an a second one in

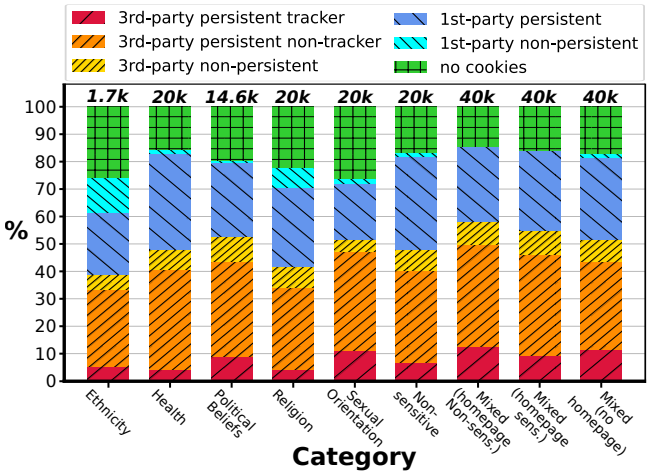

Figure 9: Cookie usage across FQDNs with homogeneous content. We group elements using the categories of Table 8 and at the top of each bar we indicate the FQDNs sampled within each category.

United Kingdom. We choose those two locations to make sure that GDPR applies and that we receive the minimum amount of cookies. After a web page has finished loading, we use the CookieCheck tool [54] to identify persistent cookies originating from third-party trackers. For both the notions of *persistence* and *tracker* we use the same definitions of [54]. The results for the analysis of cookies are shown in Figure 9. We identify fewer third-parties, around 49%, compared to the 75% reported in [54] that analyzed a set of popular web sites. We observe smaller variations in the relative percentages of third-party cookies across the sensitive categories, while in the mixed group the distribution is more uniform. Independently from the their origin, and across all the categories, web sites tend to use persistent cookies with an expiration time that exceeds one month. Around 71.5% of the sampled URLs sets at *least one persistent cookie* without user's consent. In the subset of *persistent third-party trackers*, Sexual Orientation and Political Beliefs have twice the amount of cookies than the other sensitive categories. On the other hand, we observe also some trends indicating that web site administrators with content have started taking steps to protect the privacy of their users. First, the percentage of web sites that do not set any cookie is higher for sensitive categories compared to the other ones. Second Ethnicity, Health and Religion have the lowest amount of persistent third-party trackers across all the categories. Web sites that belong to those categories use less cookies than FQDNs with mixed or non-sensitive content. We conclude that even if the amount of persistent third-party trackers appears to be smaller on some sensitive categories, still 71.5% of URLs sets persistent cookies with no prior consent from the user. More than half of such cookies originate from third-parties, and only 13% of the third-parties does not use persistent cookies.

Overall, our results make the conclusion that sensitive content is widely spread, but it is handled similarly as any other URL, without any special provision for the privacy of users. More than 30% of sensitive URLs are hosted in domains that fail to use HTTPS, and when third-party cookies are set, in 87% of the cases at least one cookie is persistent.

Table 10: The proposed solutions related to this work and their corresponding key features.

		Proprietary	Sensitive Coverage	Granularity Level
Scientific literature	This work	No	Yes	URL
	Mayers et al. [58]	No	Partial	Not available
	Wills et al. [86]	No	Partial	URL
	Razaghpanah et al. [68]	No	Partial	Mobile apps
	Carrascosa et al. [20]	No	Partial	Ads
	Iordanou et al. [43]	No	Yes	Domain
	Reyes et al. [71]	No	Partial	Mobile apps
Commercial Services	Alexa.com [11]	Yes	Partial	Domain
	SimilarWeb [74]	Yes	Partial	Domain
	McAfee LLC [53]	Yes	Partial	URL
	Symantec [78]	Yes	Partial	URL
	zvelo [91]	Yes	Partial	URLs
	cyren [31]	Yes	Partial	URLs
	Google [39]	Yes	Partial	Domain / Partial URL

5 RELATED WORK

In Section 2.1 we discussed the limitations of commercial taxonomy services (see also Table 10). In this section we focus on the small, but recently growing literature on sensitive domains, and their relationship with Web tracking. Studies like [20, 43, 58, 68, 86] are mostly about tracking, but include sections on sensitive topics usually to demonstrate that tracking happens even on such domains. These works have none of the breadth of our study. Typically they look at a limited number of hand-picked sensitive domains to detect trackers.

Some recent papers are dedicated to studying tracking in *particular types* of sensitive domains, such as pornographic sites [82], sites for minors [71] (falling under COPPA [1] jurisdiction), or in Facebook [18, 61]. Again, our main difference with these works is that they are mostly addressing the issue of *who is tracking* on such domains, whereas we are concerned with *how to find* domains of interest, and more importantly, individual URLs. To the best of our knowledge, our work is the only one devoted to developing classifiers which can detect multiple sensitive categories at the URL level. Also, the only one to construct a training-set with more than 100k sensitive URLs, and to detect sensitive domains on the entire Web instead of in a particular platform [61].

The literature on tracker detection is extensive [13–15, 34, 51, 52, 67, 70, 83]. We focus on how to find sensitive URLs in the wild, and we present only a very preliminary analysis of security and privacy issues on such web sites. Looking at who is present and what information is being collected is beyond the scope of this work and part of our future work.

Web-domain and text classification are active research areas upon which we draw tools like TF-IDF [72] and BoW [46] for feature engineering, and Naïve Bayes algorithm [9] for classification. Our contribution is more on how we combine together techniques rather on improving a specific approach. Curlie.org [30] is an ideal taxonomy for finding large lists of sensitive domains efficiently via a mix of automated and manual steps, as we did in this work. The

methodology that we presented in this paper is generic enough that such commercial labeled databases can also be used to develop classifiers to detect sensitive web sites. Our work also shows that relatively easy to implement classifiers are sufficient to identify sensitive web sites, e.g., according to the GDPR. Moreover, web site text classification increases in importance as web pages and content become increasingly dynamic. URL-based topic classification techniques that used to work well in the past [16, 17], will fail to classify dynamic web sites with sensitive content.

6 CONCLUSION

In this paper, we show how to develop a first of its kind classifier for identifying URLs that point to sensitive content according to Article 9 of GDPR. Being tracked on such sites may allow trackers to make inferences about one's health, sexual preference, political beliefs etc. Independently of the legal dimension of the matter, being able to identify such URLs programmatically in real time, opens up the road for additional proactive measures such as warning users, blocking third-parties, or even automatically filing complaints.

Training a classifier that can do this for any page on the Web is a daunting task. The training set needs to be large and diverse enough. This precludes using as training set any hand picked set of terms or web pages. Even if one could do this, ambiguities in the use of terms like Health, in both sensitive and non-sensitive contexts, would break the attempt. Instead, we used as training set a filtered subset of the largest open source taxonomy of the Web curated by human editors. This, in conjunction with careful algorithm design, feature selection, and tuning has allowed the best of our classifiers to achieve a binary classification accuracy of close to 90% and even detect individual sensitive categories with higher accuracy, e.g., Health (98%), Politics (92%), Religion (97%). We have used our classifier to search for sensitive URLs in the largest publicly available snapshot of the English speaking Web from October 2019. Our analysis of this corpus shows that a good 15.8% of the URLs are sensitive, whereas a 28% of the domains contain at least one sensitive URL. Looking at this set of domains and URLs we notice questionable practices such using HTTP instead of HTTPS, and we detect lots of persistent and third-party cookies.

In our future work we intend to analyses the identity and the methods used by third-parties on the more than 155 million sensitive URLs that we have detected. We also intend to design efficient white-/black-listing methods to avoid having to perform per-URL classification. Our initial results indicate that performing this at domain level would create lots of false positives and negatives, since many sensitive URLs are under top-level domains that seem non-sensitive (and vice versa). We also intend to examine crowdsourced approaches and federated learning techniques for distributing both the collection of (re)training sets and the (re)execution of training algorithms.

ACKNOWLEDGEMENTS

We want to thank our shepherd Amreesh Phokeer and the anonymous reviewers for their valuable comments. We would also like to thank Théo Galy-Fajou (TU Berlin) and Giovanni Cherubin (EPFL) for many useful discussions on the design, training, and evaluation of classifiers that we developed in this paper. This work was supported in part by the European Research Council (ERC) Starting Grant ResolutioNet (ERC-StG-679158) and the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No. : 871370 - PIMCity.

REFERENCES

[1] 1998. Children's Online Privacy Protection Act (COPPA). https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule.
[2] 2020. Brand Safety by Oracle. https://www.oracle.com/data-cloud/brand-safety-suitability/.
[3] 2020. Brand Safety Controls | Facebook Business Help Center. https://www.facebook.com/business/help/1926878614264962?id=1769156093197771.
[4] 2020. gensim v.3.8.1. https://pypi.org/project/gensim/3.8.1/.
[5] 2020. sklearn CountVectorizer. https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html.
[6] 2020. sklearn TfidfVectorizer. https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html.
[7] 2020. sklearn v.0.21.3. https://scikit-learn.org/stable/whats_new/v0.21.html.
[8] C. Abdelberi, T. Chen, M. Cunche, E. De Cristofaro, A. Friedman, and M. A. Kâafar. 2014. Censorship in the Wild: Analyzing Internet Filtering in Syria. In *ACM IMC*.
[9] A. B. Adetunji, J. P. Oguntoye, O. D. Fenwa, and N. O. Akande. 2018. Web Document Classification Using Naïve Bayes. *Advances in Mathematics and Computer Science* 29 (2018).
[10] G. Aggarwal, E. Bursztein, C. Jackson, and D. Boneh. 2010. An Analysis of Private Browsing Modes in Modern Browsers. In *USENIX Security*.
[11] Alexa Internet. 2019. Alexa top websites by category. https://www.alexa.com/topsites/category.
[12] L. Arras, F. Horn, G. Montavon, K-R Müller, and W. Samek. 2017. What is relevant in a text document: An interpretable machine learning approach. *PLOS One* (2017).
[13] R. Balebako, P. G. León, R. Shay, B. Ur, Y. Wang, and L. F. Cranor. 2012. Measuring the Effectiveness of Privacy Tools for Limiting Behavioral Advertising. In *W2SP Workshop*.
[14] P. Bangera and S. Gorinsky. 2017. Ads versus Regular Contents: Dissecting the Web Hosting Ecosystem. In *IFIP Networking*.
[15] M. A. Bashir, S. Arshad, W. Robertson, and C. Wilson. 2016. Tracing Information Flows Between Ad Exchanges Using Retargeted Ads. In *USENIX Security*.
[16] E. Baykan, M. Henzinger, L. Marian, and I. Weber. 2009. In *WWW*.
[17] E. Baykan, M. Henzinger, L. Marian, and I. Weber. 2011. A comprehensive study of features and algorithms for URL-based topic classification. *ACM Transactions on the Web (TWEB)* 5, 3 (2011).
[18] J. G. Cabanas, A. Cuevas, and R. Cuevas. 2018. Facebook Use of Sensitive Data for Advertising in Europe. arXiv:cs.SI/1802.05030
[19] P. Calado, M. Cristo, E. Moura, N. Ziviani, B. Ribeiro-Neto, and M. A. Gonçalves. 2003. Combining Link-based and Content-based Methods for Web Document Classification. In *ACM CIKM*.
[20] J. M. Carrascosa, J. Mikians, R. Cuevas, V. Erramilli, and N. Laoutaris. 2015. I Always Fell Like Somebody's Watching Me. Measuring Online Behavioral Advertising. In *ACM CoNEXT*.
[21] S. Chakrabarti, S. Roy, and M. V. Soundalgekar. 2003. Fast and accurate text classification via multiple linear discriminant projections. *The VLDB Journal* 12 (2003).
[22] R.-C. Chen and C.-H. Hsieh. 2006. Web page classification based on a support vector machine using a weighted vote schema. *Expert Systems with Applications* 31 (2006).
[23] H. L. Chieu and H. T. Ng. 2002. A Maximum Entropy Approach to Information Extraction from Semi-structured and Free Text. In *Proc. of the Eighteenth National Conference on Artificial Intelligence*.
[24] Common Crawl. 2018. August Crawl Archive Introduces Language Annotations. https://commoncrawl.org/2018/08/august-2018-crawl-archive-now-available/.
[25] Common Crawl. 2020. Commn Crawl. http://commoncrawl.org/.
[26] Common Crawl. 2020. So you're ready to get started. https://commoncrawl.org/the-data/get-started/.
[27] Curlie.org. 2018. How to Get Curlie Data. https://curlie.org/docs/en/help/getdata.html.
[28] Curlie.org. 2019. Any plans to resume RDF data updates? https://www.resource-zone.com/forum/t/any-plans-to-resume-rdf-data-updates-please-please.54035/.
[29] Curlie.org. 2019. Curlie - Become and Editor. https://www.curlie.org/docs/en/help/become.html.
[30] Curlie.org. 2019. Curlie - The Collector of URLs. https://curlie.org/.
[31] CYREN. 2019. Enterprise SaaS Security, Threat Intelligence Services - Cyren. https://www.cyren.com/.

[32] L. Denoyer and Gallinari P. 2004. Bayesian Network Model for Semi-structured Document Classification. *Information Processing and Management* 40 (2004).

[33] P. Domingos and M. Pazzani. 1997. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. *Machine Learning* 29 (1997).

[34] S. Englehardt and A. Narayanan. 2016. Online Tracking: A 1-million-site Measurement and Analysis. In *ACM SIGSAC*.

[35] V. Estruch, C. Ferri, J. Hernández-Orallo, and M.J. Ramírez-Quintana. 2006. Web Categorisation Using Distance-Based Decision Trees. *Electronic Notes in Theoretical Computer Science* 157 (2006).

[36] European Commission. 2018. Art. 9 GDPRProcessing of special categories of personal data. https://gdpr-info.eu/art-9-gdpr/.

[37] European Commission. 2018. Data protection in the EU, The General Data Protection Regulation (GDPR); Regulation (EU) 2016/679. https://ec.europa.eu/info/law/law-topic/data-protection/.

[38] X. Gao, Y. Yang, H. Fu, J. Lindqvist, and Y. Wang. 2014. Private Browsing: An Inquiry on Usability and Privacy Protection. In *WPES*.

[39] Google. 2019. Google Ads - About Display Planner. https://support.google.com/google-ads/answer/3056432?hl=en.

[40] S. Greengard. 2018. Weighing the Impact of GDPR. *Comm. of the ACM* 61, 11 (2018).

[41] E. H. Han, G. Karypis, and V. Kumar. 2001. Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification. In *Proc. of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining*.

[42] L. Howard, P. Liam, B. Yevgen, and X. Y. Simon. 2010. Document Classification Using Information Theory And A Fast Back-Propagation Neural Network. *Intelligent Automation and Soft Computing* 16 (2010).

[43] C. Iordanou, G. Smaragdakis, I. Poese, and N. Laoutaris. 2018. Tracing Cross Border Web Tracking. In *ACM IMC*.

[44] L. Kalman. 2019. New European Data Privacy and Cyber Security Laws: One Year Later. *Comm. of the ACM* 62, 4 (2019).

[45] S. Khattak, M. Javed, S. A. Khayam, Z. A. Uzmi, and V. Paxson. 2014. A Look at the Consequences of Internet Censorship Through an ISP Lens. In *ACM IMC*.

[46] Y. Ko. 2012. A Study of Term Weighting Schemes Using Class Information for Text Classification. In *ACM SIGIR*.

[47] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. E. Barnes, and D. E. Brown. 2019. Text Classification Algorithms: A Survey. *Information* 10, 4 (2019).

[48] G. Krishnaveni and T. Sudha. 2016. Naïve Bayes Text Classification – A Comparison of Event Models. *Imperial Journal of Interdisciplinary Research* 3 (2016).

[49] O. Kwon and L. Jong-Hyeok. 2003. Text categorization based on k-nearest neighbor approach for Web site classification. *Information Processing and Management* 39 (2003).

[50] Q. Le and T. Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*.

[51] A. Lerner, A. Kornfeld Simpson, T. Kohno, and F. Roesner. 2016. Internet Jones and the Raiders of the Lost Trackers: An Archaeological Study of Web Tracking from 1996 to 2016. In *USENIX Security*.

[52] C. Leung, J. Ren, D. Choffnes, and C. Wilson. 2016. Should You Use the App for That?: Comparing the Privacy Implications of App- and Web-based Online Services. In *ACM IMC*.

[53] McAfee LLC. 2020. Customer URL Ticketing System. https://www.trustedsource.org/.

[54] M. Trevisan and S. Traverso and E. Bassi and M. Mellia. 2019. 4 Years of EU Cookie Law: Results and Lessons Learned. *PoPETs* (2019).

[55] KM. Mahesh, DH. Saroja, GD. Prashant, and C. Niranjah. 2015. Text mining approach to classify technical research documents using naïvebayes. *International Journal of Advanced Research in Computer and Communication Engineering* 4 (2015).

[56] L. Manevitz and M. Yousef. 2007. One-class document classification via Neural Networks. *Neurocomputing* 70 (2007).

[57] S. Matic, G. Tyson, and G. Stringhini. 2019. PYTHIA: a Framework for the Automated Analysis of Web Hosting Environments. In *WWW*.

[58] J. R. Mayer and J. C. Mitchell. 2012. Third-Party Web Tracking: Policy and Technology. In *IEEE Symposium on Security and Privacy*.

[59] Y. Meng, J. Shen, C. Zhang, and J. Han. 2018. Weakly-Supervised Neural Text Classification. In *ACM CIKM*.

[60] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*.

[61] J. G. Caba nas, A. Cuevas, and R. Cuevas. 2018. Unveiling and Quantifying Facebook Exploitation of Sensitive Personal Data for Advertising Purposes. In *USENIX Security Symposium*.

[62] Office of the Australian Information Commissioner. 2018. Australian Privacy Principles guidelines; Australian Privacy Principle 5 – Notification of the collection of personal information. https://www.oaic.gov.au/agencies-and-organisations/app-guidelines/.

[63] Office of the Privacy Commissioner of Canada. 2018. Amended Act on The Personal Information Protection and Electronic Documents Act. https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/.

[64] K. Oppenheimer. 2015. Topical web-page classification. MSc thesis, Computer Science, Brandeis University, https://github.com/kahliloppenheimer/Web-page-classification/blob/master/paper.pdf.

[65] Personal Information Protection Commission, Japan. 2017. Amended Act on the Protection of Personal Information. https://www.ppc.go.jp/en/.

[66] V. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *NDSS*.

[67] E. Pujol, O. Hohlfeld, and A. Feldmann. 2015. Annoyed Users: Ads and Ad-Block Usage in the Wild. In *IMC*.

[68] A. Razaghpanah, R. Nithyanand, N. Vallina-Rodriguez, S. Sundaresan, M. Allman, C. Kreibich, and P. Gill. 2018. Apps, Trackers, Privacy, and Regulators: A Global Study of the Mobile Tracking Ecosystem. In *NDSS*.

[69] A. Razaghpanah, R. Nithyanand, N. Vallina-Rodriguez, S. Sundaresan, M. Allman, C. Kreibich, and P. Gill. 2019. Apps, Trackers, Privacy, and Regulators: A Global Study of the Mobile Tracking Ecosystem. In *NDSS*.

[70] B. Reuben, L. Ulrik, M. Van Kleek, J. Zhao, T. Libert, and N. Shadbolt. 2018. Third Party Tracking in the Mobile Ecosystem. *CoRR* (2018).

[71] I. Reyes, P. Wijesekera, J. Reardon, A. Elazari, A. Razaghpanah, N. Vallina-Rodriguez, and S. Egelman. 2018. "Won't Somebody Think of the Children?" Examining COPPA Compliance at Scale. (2018).

[72] G. Salton and C. Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing and Management* 24 (1988).

[73] I. Sánchez-Rola, M. Dell'Amico, P. Kotzias, D. Balzarotti, L. Bilge, P-A. Vervier, and I. Santos. 2019. Can I Opt Out Yet?: GDPR and the Global Illusion of Cookie Control. In *ASIACCS*.

[74] SimilarWeb. 2019. SimilarWeb - Top sites ranking for all categories in the world. https://www.similarweb.com/top-websites.

[75] K. Solomos, P. Ilia, S. Ioannidis, and N. Kourtellis. 2019. TALON: An Automated Framework for Cross-Device Tracking Detection. In *RAID*.

[76] State of California. 2018. California Consumer Privacy Act – Assembly Bill No. 375. https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375.

[77] A. Sun, E.-P. Lim, and W.-K. Ng. 2002. Web Classification Using Support Vector Machine. In *Proc. of the 4th International Workshop on Web Information and Data Management*.

[78] Symantec. 2018. Symantec RuleSpace: OEM URL Categorization Database and Real-Time Web Categorization Technology. https://www.symantec.com/products/rulespace.

[79] The Privacy Protection Authority of Israel. 2018. Protection of privacy regulations (data security) 5777-2017. https://www.gov.il/en/Departments/legalInfo/data_security_regulation.

[80] Y. Tian, T. Huang, W. Gao, J. Cheng, and P. Kang. 2003. Two-phase Web site classification based on hidden Markov tree models. In *Proc. IEEE/WIC International Conference on Web Intelligence*.

[81] S. L. Ting, W. H. Ip, and A. H. C. Tsang. 2011. Is Naïve Bayes a Good Classifier for Document Classification? *International Journal of Software Engineering and Its Applications* 5 (2011).

[82] P. Vallina, A. Feal, J. Gamba, N. Vallina-Rodriguez, and A. F. Anta. 2019. Tales from the Porn: A Comprehensive Privacy Analysis of the Web Porn Ecosystem. In *ACM IMC*.

[83] N. Vallina-Rodriguez, J. Shah, A. Finamore, Y. Grunenberger, K. Papagiannaki, H. Haddadi, and J. Crowcroft. 2012. Breaking for commercials: Characterizing mobile advertising. In *ACM IMC*.

[84] W3C. 2014. RDF - Semantic Web Standards. https://www.w3.org/RDF/.

[85] F. Wang, Q. Wang, N. Feiping, Y. Weizhong, and W. Rong. 2018. Efficient tree classifiers for large scale datasets. *Neurocomputing* 284 (2018).

[86] C. E. Wills and C. Tatar. 2012. Understanding what they do with what they know. In *WPES*.

[87] Y. Wu, P. Gupta, M. Wei, Y. Acar, S. Fahl, and B. Ur. 2018. Your Secrets Are Safe: How Browsers' Explanations Impact Misconceptions About Private Browsing Mode. In *WWW*.

[88] www.odp.org. 2019. Open Directory Project.org: ODP Web Directory Built With the DMOZ RDF Database. https://www.odp.org/.

[89] Y. Yao and Z. Xiao and B. Wang and B. Viswanath and H. Zheng and B. Y. Zhao. 2017. Complexity vs. Performance: Empirical Analysis of Machine Learning as a Service. In *IMC*.

[90] D. Zhang and W. S. Lee. 2004. Web Taxonomy Integration Using Support Vector Machines. In *WWW*.

[91] zvelo. 2019. Check a URL Category | URL Database For DNS/IP & Web Filtering. https://tools.zvelo.com/.

APPENDIX

A EXTRACTING SENSITIVE URLS FROM UNKNOWN ELEMENTS

In order to include more samples in each category we first classify all the *Unknown* elements using the baseline classifier. Since the overall number of categories in our classifier is six, the minimum prediction probability needed in order to assign a class for a given URL is ≈ 0.17, assuming that the URL content is diverse enough to cover all the categories. Nevertheless, if a given URL shows biases towards only two classes the prediction probability can climb up to 0.5. In order to avoid polluting our dataset with mis-classified URLs we set a minimum threshold on the prediction probability for all sensitive categories to be above 0.5. Note that during our analysis the lower prediction probability that we observe for the sensitive categories was equal among all of them at ≈ 0.3.

In the case of the Non-sensitive category, we include the top 157,118 URLs to balance the ratio between sensitive and non-sensitive URLs based on their prediction probability. In this case, since the number of URLs belonging to the Non-sensitive category is bounded by the total number of URLs that we need for balancing the ratio between sensitive and non-sensitive, the lower prediction probability that we observe for the included URLs was 0.97.

The second column of Table 6 reports the new elements that are added to each category. Despite our attempt to keep the categories balanced, we were unable to achieve an equal split among the five sensitive categories. On one side, for extremely generic categories such as Ethnicity, we had hard times finding those elements even in our manually validated ground truth. On the other side, Curlie is a community-driven project focused on detecting a wide range of different topics, it might simply lack samples for specific categories (e.g., Sexual Orientation). The column "Final" shows the overall number of URLs that were included in this expanded version of our training dataset that we use to build the balanced classifier.

To confirm that the samples that were added to each category are correctly classified, we sampled 100 elements from each category and manually validated the categorization. Our investigation shows that more than 90% of the elements were correctly labeled.