The International Journal of Parallel, Emergent and Distributed Systems
Vol. 00, No. 00, Month 2011, 1–22

RESEARCH ARTICLE

On web user tracking of browsing patterns for personalised advertising

Silvia Puglisi*, David Rebollo-Monedero and Jordi Forné[a]

*Corresponding author. Email: silvia.puglisi@upc.edu

[a]*Department of Telematics Engineering,*
Universitat Politècnica de Catalunya (UPC)
C. Jordi Girona 1-3, 08034 Barcelona, Spain;

(*v3.6 released March 2011*)

On today's Web, users trade access to their private data for content and services. App and service providers want to know everything they can about their users, in order to improve their product experience. Also, advertising sustains the business model of many websites and applications. Efficient and successful advertising relies on predicting users' actions and tastes to suggest a range of products to buy. Both service providers and advertisers try to track users' behaviour across their product network. For application providers this means tracking users' actions within their platform. For third-party services *following* users, means being able to track them across different websites and applications. It is well known how, while surfing the Web, users leave traces regarding their identity in the form of activity patterns and unstructured data. These data constitute what is called the user's online footprint. We analyse how advertising networks build and collect users footprints and how the suggested advertising reacts to changes in the user behaviour.

Keywords: privacy; ubiquitous-tracking; privacy metrics; advertising;

1. Introduction

Websites use *personalisation services* to provide a tailored experience to their visitors. In order to make their product more personal to the single users they need to keep profiles of their users, collect their in page reading activities and eventually their preferences. This data is then shared to third-party services, accessed and analysed without users? direct consent. Furthermore, records of users? activities are used for different purposes, most unknown to the end user, such as marketing or to provide analytics services back to the original website or application. Among the data analysed by websites are also included user preferences and social connections. These can be obtained by tracking users across different applications and sites through cookies or open web sessions. Even if the user does not accept cookies or is not logged into a service account, such as their Google, Twitter or Facebook accounts, the web page and third-party services can still try to profile them by using third-party http requests, among other techniques. Within the http request various selectors can be included to communicate user preferences or particular features, in the form of URL variables. Features that might be used by advertising networks and malicious trackers include personalised language or fonts

ISSN: 1744-5760 print/ISSN 1744-5779 online

DOI: 10.1080/17445760.YYYY.CATSid
http://www.informaworld.com

settings, browser extensions, in page keywords, battery charge and status, and so on. These features are then used to identify individual users by restricting the pool of possible candidates among all the visitors in a certain time frame, location, profile of interests. Unique users can then be distinguished across multiple devices or sessions.

1.1 Contribution

We have observed how users are tracked across the Web and how the displayed advertising is tailored even after they have visited a few websites with a certain interest bias. In previous work [1] [2] we analysed how third-party advertising services are able to profile users on a short series of websites visited and how these are able to *follow* users while they surf the web. In our study we analyse how the user profile detected by advertising services can be used to estimate the user privacy risk on a certain network. We analyse how advertising networks identify a user and start tracking them, by considering keywords contained in the webpage and understanding the underlying network structure of tracked domains. We measure the distance between the observed user profile and the actual user profile, by categorising the set of keywords contained in web pages and by capturing third-party http requests. We introduce a set of metrics to express this distance between the two profiles.

It is important to note that we have considered the case for which users are not registering, neither connecting any external account, as it could be the case with services like: Facebook, Google+, Twitter, and so on. In such scenario we have measured how these networks still attempt to track the user by sending user information through http requests to their services.

The main contributions of this paper are the following.

(1) We present a model of the user profile that is able to capture how each website and tracking network categorise their activities in terms of interests and interactions.
(2) We analyse how much information is sent by each page visited to third-party services by measuring the partial user profile and the actual user profile. The partial user profile is what the website and third-party services know about the user. The actual user profile is instead the full profile measured at the end of the series of page visited.
(3) We introduce a set of metrics to express the relationship between the partial and the actual user profile.
(4) We profile third-party http calls sent by Facebook tracking services and compare this to the the user actual profile.
(5) We model user online footprints as a graph of the actions generated by each user and analyse the resulting graph structure, identifying known malicious trackers.

2. State of the Art

Information regarding locations, browsing habits, communication records, health information, financial information, and general preferences regarding user online and offline activities are shared by different parties online. This level of access is often directly granted from the user of such services. In a wide number of occasion though, private information is captured by online services without the direct user consent or even knowledge. We believe that the privacy and sensitiveness of the

information becoming accessible to third parties can be easily overlooked.

Personal computers and more generally communication devices that are carried around by people are capable of being located, identified and tracked across different locations, networks and services [3]. All these devices can therefore be used for a variety of surveillance activities, which are in itself detrimental to the user's interests. Until recently in fact, the cost of surveillance and tracking of people and activities was proportional to the cost of directly reaching, asking or following a single person or a group of people. Technology therefore enhances the surveillance capabilities by introducing tools that allow the collection of information arising from a person's activities. This information can furthermore be combined and inferred, therefore offering a more complete picture of that person.

For example, to personalise their services or offer tailored advertising, web applications could use tracking services that identify a user through different networks [4] [5]. These tracking services usually combine information from different profiles that users create, for example their Gmail address or their Facebook or LinkedIn accounts. In addition, specific characteristics of the user's device can be used to identify them through different sessions and websites, as described by the Panopticlick project [6].

Browser fingerprinting is a technique implemented by analytics services and tracking technologies to identify uniquely a user while they browser different websites. Different features of a specific browser setup can be used to identify uniquely a user. Supported languages, browser extensions or installed fonts [7] can be used to identify a browser setup among others. More advanced techniques distinguish between browsers' JavaScript execution characteristics [8]. These features are particularly interesting since they are more difficult to simulate or mitigate in practice. Targeting JavaScript execution characteristics actually means looking at the innate performance signature of each browser's JavaScript engine, allowing the detection of browser version, operating system and microarchitecture. These attacks can also work in situations where traditional forms of system identification (such as the user-agent header) are modified or hidden. Other techniques exploit the whitelist mechanism of the popular NoScript Firefox extension. This mechanism allows the user to selectively enabling web pages' scripting privileges to increase privacy by allowing a site to determine if particular domains exist in a user's NoScript whitelist.

It is important to note that while tracking creates serious privacy concerns for internet users, the customisation of results is also beneficial to the end user [9]. In fact, while tailored services offer to the user only information relevant to their interests, it also allows some companies and institutions to concentrate an enormous amount of information about internet users in general. [10] investigate user profiling and access mechanisms offered by online data aggregator to users' collected data. Both the collected data and its accuracy was analysed together with the user's concerns. In their findings about 70% of the participants to the study expressed some concerns about the collection of sensitive data, its level of detail and how it might be used by third parties, especially for credit and health information.

Generally speaking, the activity of tracking a user across different websites, visits and devices, involves three main actors: the user, the tracking network, the list of websites visited. Every time a user visits a website a piece of code on the page is called asynchronously from the user's browser. When the call to the tracing network is performed a number of user data is transferred and used to profile the user at a later time and/or on a different website or device. By modelling the user behaviour as a directed graph, it is possible to uncover the underling network structure of the user footprint and the tracking networks tracing the user across the web [11] [12].

It has been shown how most successful tracking networks exhibit a consistent

structure across markets, with a dominant connected component that, on average, includes 92.8% of network vertices and 99.8% of the connecting edges [13]. [13] have measured the chance that a user will become tracked by all top 10 trackers in approximately 30 clicks on search results to be of 99.5%. More interesting, [13] have shown how tracking networks present properties of the small world networks. Therefore, implying a high-level global and local efficiency in spreading the user information and delivering targeted ads.

An interesting property of networks to understand their architecture is the behaviour of the average degree of nearest neighbours [14] [15]. The average degree of the nearest neighbours of a node $k_{nn}(k)$ is a quantity related to the correlations between the degree of connected vertices [16], since it can be expressed as the conditional probability that a given vertex with degree k is connected to a vertex of degree k'. This property defines if the network in consideration is assortative, if k_{nn} is an increasing function of k or dissortative [17] if it is not. The property of assortativity has been used in the field of epidemiology, to help understand how a disease or cure spreads across a network. It is particularly interesting to note that assortativity can give a measurement if the removal of a set of network's vertices may correspond in curing, vaccinating or quarantining individual cells in the network.

Another interesting aspect of networks is the presence of *communities*. A common activity when analysing large network is to start finding communities by dividing the nodes into *modules*. A common approach applies *generative models* able to infer the model parameters directly from the data. A simple generative process is the Stochastic Block Model (SBM) [18]. A stochastic block model is able to explicitly describe the global structure of a network, providing a model of how the network can be partitioned into subgroups (blocks) and how the probability distribution of the connections between the nodes (i.e. probability that a node is connected to another) depends on the blocks to which the nodes belong [19].

The microcanonical formulation [20] of blockmodels takes as parameters the partition of the nodes into groups b and a $B \times B$ matrix of edge counts e, where e_{rs} is the number of edges between groups r and s. Since edges are then placed randomly, nodes belonging to the same group possess the same probability of being connected with other nodes of the network. Furthermore, to be able to find small groups in large network nested SBM are used. With nested SBM groups are clustered into groups, and the matrix e of edge counts are generated recursively from another SBM [21]. Agglomerative multilevel Markov chain Monte Carlo (MCMC) algorithm as described in [22], can be implied to compute a partition of the resulting graph.

Protection techniques against tracking networks are implemented through software agents able to identify if third-party requests are accessing private data. These agents include Privacy Badger [23], Mozilla Lightbeam [24], Ghostery [25], AdBlock [26], and so on. Some of these agents block certain Javascript functions, or attempts to access determined browser functionalities that can be used to uniquely identify the user. Some others implement a Tracking Protection Lists (TPL). A TLP can be seen as a blacklist of identified tracking domains that user might want to block.

Another interesting aspect of advertising services is how they are designed to work on feedback loops [27]. An advertising service can in fact be seen as a black-box providing the tracker trying to identify or profile the user, and the returned advertising content. The tracker is used to send information back to the advertising service, which in response will return a certain content tailored to the user preferences. Within this feedback loop different aspect of the user behaviour are taken in consideration. These include certainly the users browsing history and their click

through rate, i.e. a measurement of the amount of time users in a population are more likely to interact with an ad. In more sophisticated advertising solution also user social connections are taken in consideration.

Advertising therefore services raise the problem of confidentiality of the user reading activity [28]. Up to know an eloquent example of this problem was provided by the way public library in the US operates. Reading activities were considered historically private and were protected through a set of rules that restricted libraries ability to exploit reading records. This regime is clearly bypassed when libraries decide to provide digital services to their users. Digital services providers and third parties can in fact access users reading activities without agreeing to the library confidentiality regime.

3. Modelling the user profile

Each time the user visits a new page, we aggregate the page keywords and build what we consider the user's profile of interests (Figure 4). We consider a tractable model of the user profile as a probability mass function (PMF), as proposed in [29, 30], to express how each keyword contributes to expose how many times the user has indirectly expressed a preference toward a specific category. We consider that the user expresses a preference when they visit a webpage categorised with certain keywords. This model follows the intuitive assumption that a particular category is weighted according to the number of times this has been counted in the user profile.

We define the profile of a user as the PMF $p = (p_1, \ldots, p_L)$, conceptually a histogram of relative frequencies of tags across the set of tag categories $\mathcal{T}$. This means that we group tags around interests using top level categories as defined by the Open Directory Project (DMOZ) [31]. The user profile is calculated at the end of the series of website visited by the user. Similarly we define the partial user profile at moment as this is known to the advertising network as $\hat{p} = (\hat{p}_1, \ldots, \hat{p}_L)$.

Note that, for the case when an advertising network is present on each and every page, $\hat{p} = p$ at the end of the series of sites visited. This means that the network was able to record each page visited by the user. This could easily be the page of advertising networks like Google that through different third-party services are ubiquitously present across the web.

We also define the profile of an ad, or third-party http request as the PMF $q = (q_1, \ldots, q_L)$, where q_l is the percentage of tags belonging to the category l which have been assigned to this specific advertising item. You can think of the ad profile as the PMF of the tag contained in every http request sent from the visited page to the advertising network (Listings: 1, 2, 3). This profile notes which tags the tracking network is using to identify the user and display some advertising content. Note that the ads profile, is calculated independently for each advertising network.

Both user and ads profiles can then be seen as normalised histograms of tags across categories of interest. Our profile model is in this extent equivalent to the tag clouds that numerous collaborative tagging services use to visualise which tags are being posted, collaboratively or individually by each user. A tag cloud, similarly to a histogram, is a visual representation in which tags are weighted according to their relevance.

In view of the assumptions described in the previous section, our privacy attacker boils down to an entity that aims to profile users by representing their interests in the form of normalised histograms, on the basis of a given categorisation.

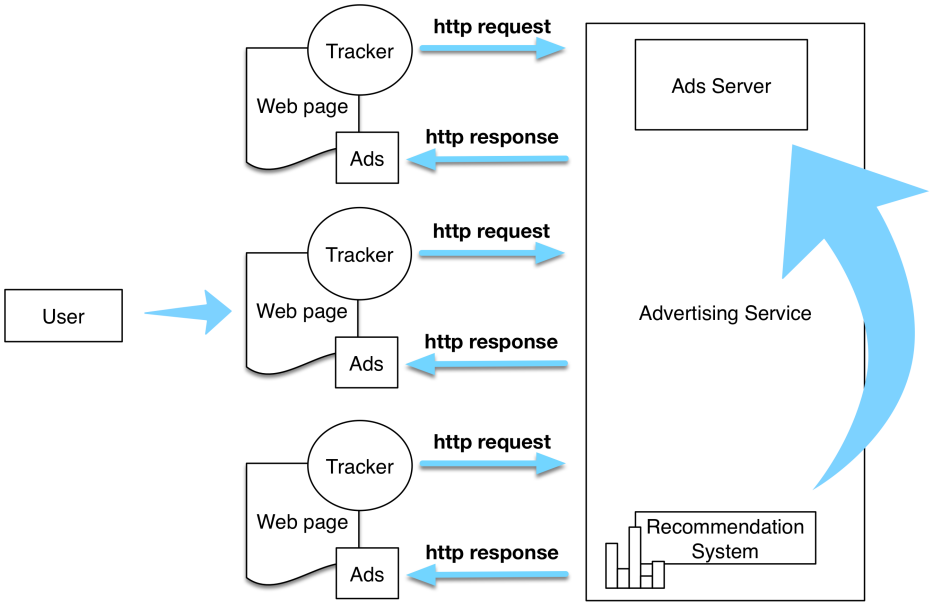

Figure 1.: Advertising services work in a feedback loop. The image illustrate how while a user surf a number of web pages, the service record their profile and adapts the returned advertising.

3.1 Tracking metrics

We consider the third-party advertising network to operate like a recommendation system that suggest products or services that might be of interest for the user, based on their preferences. A recommendation system can be described as an information filtering system that seeks to predict if the user is interested or not in a particular resource. We assume that the ad server suggests advertising based on a measure of *similarity* between what the user *does* and what the network *knows*. Furthermore, we consider tracking service to work in a feedback loop (Figure 1). When a user surfs the web each tracker on the visited pages communicates with the advertising service, sending a number of parameters through http requests. These contain the user preferences and browsing history which will be taken into consideration when ads are returned to display on the page.

It is important to note that while it is safe to consider an advertising network as a recommendation system, we should also consider that a number of processes and interactions between the advertising networks, the website, and the ultimate advertiser, can influence the actual recommendation that is displayed to the user. Tracking services can in fact follow different strategies to recommend products to users. Some services display in page advertising where a bidding mechanism allow advertisers to compete for categories and spaces, other services might decide to target only specific categories, others might instead decide to target the visited page only.

We measure the user profile, as previously described, as a histogram of their recorded preferences, and the advertising profile as a histogram of the ads that the user has received. We have considered a set of metric to measure how the advertising network is tracking the user profile, and how a page sends information to a tracking service by transmitting a partial user profile.

In previous works[1] [2] we used the *1-norm*, *2-norm* as measures of how the advertising profile, or the partial user profile, approximates the user profile. Please recall that the partial user profile is calculated by a given advertising network at a given moment on a series of pages visited.

We now introduce the normalised α-*norm* as the generalised variation GV between two probability distributions, the partial and the genuine user profiles. Furthermore, we will introduce the *KL-divergence* as a measure of how the partial profile approaches the genuine user profile. Please note that while we are defining our metrics between the partial user profile and the genuine user profile, the same assumptions holds, without loss of generality, if we compare the user's and the advertising profiles.

3.1.1 Norm and generalised variation

We define the $\alpha - norm$ between the partial profile as observed by an advertising platform and the genuine user profile as:

$$\|p - \hat{p}\|_\alpha = \sqrt[\alpha]{\sum_l |p_l - \hat{p}_l|^\alpha} \quad \text{with} \quad \alpha \in [0, \infty]$$

The case for $\alpha = \infty$ is defined in the limit:

$$\lim_{\alpha\to\infty} \|p - \hat{p}\|_\alpha = \lim_{\alpha\to\infty} \sqrt[\alpha]{\sum_l |p_l - \hat{p}_l|^\alpha} \approx \max_l |p_l - \hat{p}_l|$$

The $\alpha - norm$ is a norm in $\mathbb{R}^L$ with the following properties:

- Absolute homogeneity.
- Positive definiteness: $\|p - \hat{p}\|_\alpha \leq 0$ with equality if $p = \hat{p}$ and $\|p - \hat{p}\|_\alpha = \sqrt[\alpha]{2} \Leftrightarrow$ p and $\hat{p}$ are orthonormal deltas.
- Triangle inequality.

For $\alpha = 1$ we define the *1-norm* between the partial and the genuine user profiles as:

$$\|p - \hat{p}\|_1 = \sum_l |p_l - \hat{p}_l|$$

The *1-norm* represent the average discrepancy between the two profiles. For $\alpha = 2$ we define the *2-norm* as:

$$\|p - \hat{p}\|_2 = \sqrt{\sum_l |p_l - \hat{p}_l|^2}$$

The *2-norm* represents the Euclidean distance between the two distributions. When considering the *2-norm* it is possible to highlight larger discrepancies on the set of categories analysed. An interesting property is that *norms* are also nested. Hence:

$$\|p - \hat{p}\|_\infty \leqslant \|p - \hat{p}\|_2 \leqslant \|p - \hat{p}\|_1$$

We hence define the generalised variation $\mathrm{GV}(p, \hat{p})$, based on α-norm as:

$$\mathrm{GV}(p,\hat{p})_\alpha = \frac{1}{\sqrt[\alpha]{2}}\|p-\hat{p}\|_\alpha = \frac{1}{\sqrt[\alpha]{2}}\sqrt[\alpha]{\sum_l (p_l-\hat{p}_l)^\alpha}, \quad \alpha \in [1,\infty]$$

The coefficient $\frac{1}{\sqrt[\alpha]{2}}$ normalises the range of values of the $\alpha - norm$ in $[0,1]$. Therefore, $\mathrm{GV}(p,\hat{p})$ is a norm, is positive definite, absolutely homogeneous and satisfies the triangle inequality:

- $\mathrm{GV}(p,\hat{p}) \geq 0$ with equality if and only if $p = \hat{p}$
- $\mathrm{GV}(p,\hat{p}) \leq 1$ with equality if and only if p and $\hat{p}$ are orthonormal canonical vectors (discrete deltas).

Note that for $\alpha = 1$ the generalised variation $\mathrm{GV}(p,\hat{p})$ is equal to the total variation $\mathrm{TV}(p,\hat{p})$:

$$\mathrm{TV}(p,\hat{p}) = \frac{1}{2}\|p-\hat{p}\|_1 = \frac{1}{2}\sum_l |p_l-\hat{p}_l|$$

For $\alpha = 2$ we have the normalised *2-norm*:

$$\mathrm{GV}_2 = \frac{1}{\sqrt{2}}\|p-\hat{p}\|_2 = \frac{1}{\sqrt{2}}\sqrt{\sum_l |p_l-\hat{p}_l|^2}$$

Finally, the case for $\alpha = \infty$, $\lim_{\alpha\to\infty} \mathrm{GV}(p,\hat{p})_\alpha = \|p-\hat{p}\|_\infty$ between p and $\hat{p}$. The reason is that for $\alpha \gg 1$ the greatest difference dominates in the sum $\sum_l |p_l-\hat{p}_l|^\alpha$. Therefore $\lim_{\alpha\to\infty}\|p-\hat{p}\|_\alpha \approx \max_l |p_l-\hat{p}_l|$.

One might interpret these norms as $\alpha = 1$ been an average-case metric, $\alpha = \infty$ being a worst-case scenario, and $\alpha = 2$ a robust middle ground.

3.1.2 KL-Divergence

Now we propose and justify an information-theoretic quantity as a measure of how the partial profile approaches the genuine user profile: the *KL-divergence*. Suppose that we might interpret the profile $\hat{p}$ observed by a third-party tracking service, as a sequence of L independent, identically distributed, drawings of a user's genuine profile of interest p. Then in accordance with the rationale proposed in [32] [33], we may argue that the probability $p(\hat{p})$, of a given observed profile is related to the KL-divergence between the empirical observation $\hat{p}$ and the ideal one p, as follows:

$$-\frac{1}{L}\log p(\hat{p}) \xrightarrow[L\to\infty]{} \mathrm{D}(\hat{p}\|p)$$

Informally this means $p(\hat{p}) \approx 2^{-\mathrm{L}\ \mathrm{D}(\hat{p}\|p)}$. Note that small divergences will lead to likely outcomes, whereas large divergence associate with rare events.

Note also that $\hat{p}$ is absolutely continuous with respect to p: $p_l = 0 \Rightarrow \hat{p}_l = 0$. Also $\hat{p} \ll p \iff \mathrm{D}(\hat{p}\|p) < \infty$.

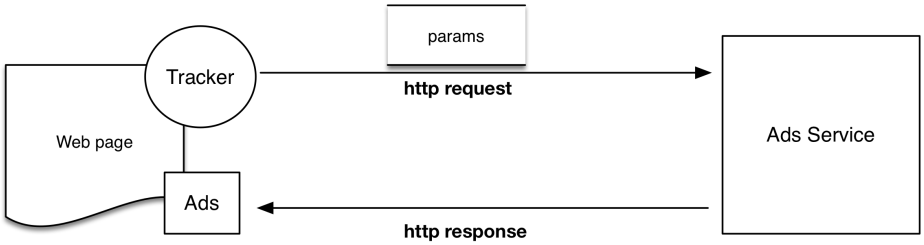

Figure 2.: Trackers on web pages make third-party http requests to advertising services. These return ads content tailored to the user web history or expressed preferences.

4. Modelling the user's online footprint

We model the user's activity as series of events belonging to a certain identity. Each event is a document containing different information. An event correspond to an action generated by the user or one of their devices. When a user visits a website or creates a post on a blog, an event is created. We can think of an event as a hypermedia document i.e. an object possibly containing graphics, audio, video, plain text and hyperlinks. We call the hyperlinks selectors and we use these to build the connections between the user's different identities or events. Each identity can be a profile or account that the user has created onto a service or platform, or just a collection of events, revealing something about the user. With account we mean an application account or a social network account, such as their LinkedIn or Facebook unique IDs. When the user visit a web page, or uses a web or mobile application, a series of events is generated and associated with the account. Some of these events are created by direct user's actions, others are created by code triggered indirectly by the user.

While the user visits a webpage and reads its content a series of snippets of code and client side scripts are executed and information is transmitted to the page backend or some third-party server. Among the information transferred are a number of user preferences. These can be their geographical location, battery level of their current used device, browser preferences, or just their browsing history captured up to that point. Some or all of the meta and in page keywords used to describe the page are also transferred. We build the user profile by collecting the meta keywords expressed in web pages. We consider this a subset of the possible set of preferences that third-party advertising networks might be interested in collecting.

4.1 Proposing a model of third-party requests on web pages

When a user visits a web page, the browser sends an http request to the server to request a representation of the resource described through the url. The server provides the resource representation in the form of a html document and the browser parses it. The html document contains a number of links to other resources, such as JavaScript code, videos, audios or images (Figure: 2). Some of these can be stored on the same domain as the requested page, some may be requested to third-party services. Such is the case of services like Google Analytics, share buttons from different social networks, or advertising banners. Together with the http request, a number of parameters are included. These contains keywords, users? preferences, information regarding the user device and session, in page information sent to the third-party service from the website or application.

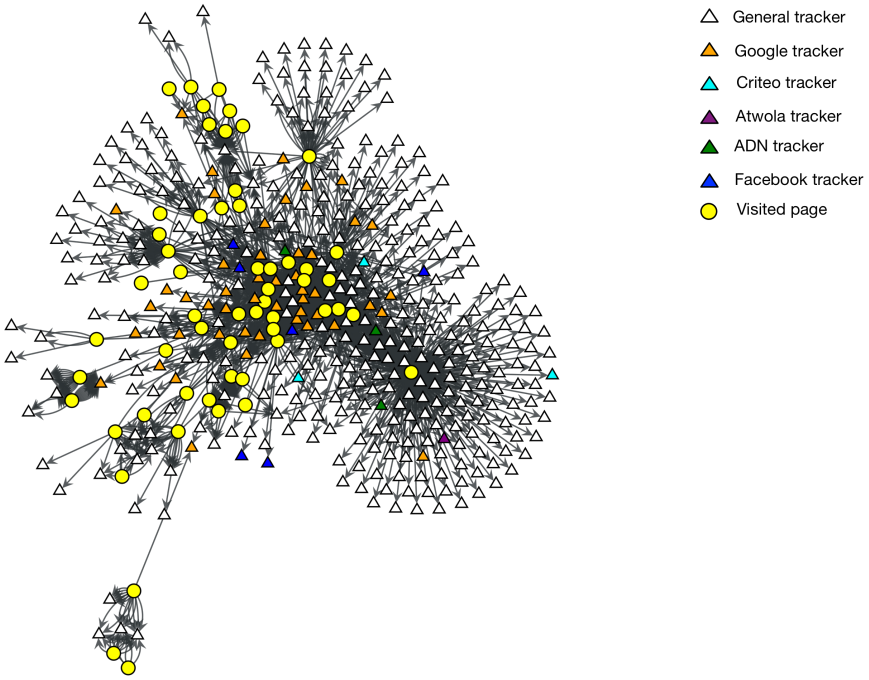

Figure 3.: The graph shows how known trackers are connected to visited pages and therefore how these are able to follow users across different websites.

When a third-party request is performed by the visited page, we store the parameters passed and if the call belongs to a known tracking network we categorise the corresponding keywords. Also when a request is made, we store a direct link between the page and a tracking domain, such as google.com. This results in a graph model of tracking networks and how these are connected to pages (Figure: 3). The graph model allows us to understand the underling network structure of tracking networks and how these are pervasively following users across their visits. In fact, every time we discover which tracking services are active on a certain website, we can create an indirect link between the user and the tracker.

4.2 Network structure metrics

We said that advertising networks or privacy attackers need to be able to *follow* the user across as many websites as possible in order to profile their interests. This naturally translates onto a graph model where each page is directly connected to its active trackers (Figure: 3). We therefore considered a set of metrics that can uncover the underlying network structure of tracking service. The first of the metrics considered is the average degree of the neighbourhood. The average degree of the neighbourhood of each node is a good indication of how many pages are connected to a certain advertising service or tracking domain.

The average degree of the neighbourhood of a node i is calculated as:

$$\langle k_{nn,i} \rangle = \frac{1}{|N(i)|} \sum_{j \in N(i)} k_j$$

Where $N(i)$ are the neighbours of node i and k_j is the degree of node j which belongs to $N(i)$.

If a certain tracker domain is connected to the majority of the page visited by a certain user, this means that they have been able to collect the user's preferences and reading activities across a number of websites. The more a tracker domain is connected, the more the user might consider this a *risk* for their privacy. We used the average degree of the neighbourhood of a tracker to rank tracker domains.

To describe the resulting network structure we also calculated the average scalar assortativity coefficient [17] defined as:

$$r = \frac{\sum_{xy} xy(e_{xy} - a_x b_y)}{\sigma_a \sigma_b}$$

Where $a_x = \sum_y e_{xy}$ and $b_y = \sum_x e_{xy}$, and e_{xy} is the fraction of edges from a vertex of type x to a vertex of type y.

We also generated a partition of SBM and nested SBM of the resulting graph employing an agglomerative multilevel Markov chain Monte Carlo (MCMC) algorithm as described in [22] [34][20]. The idea behind using SBM to describe the network structure of identified trackers is to be able to identify similar trackers and to understand if trackers belonging to the same domain or that exhibit similar behaviour can be grouped based on network properties.

5. Experimental methodology and results

We analysed the browsing habits of 50 users of Twitter, by observing the set of websites links shared for each of the top categories from the Open Directory Project (DMOZ) [31] for a total of 100 pages per user. We assumed that the articles shared on twitter are a subset of the website that each user visit every day. More importantly if they are active Twitter users, these websites will express their interest bias towards certain categories. To validate our strategy, we observed that Twitter itself offer website owners the possibility to track conversions on their pages coming from tweets and twitter ads. Please note that the list of links was only considered as a list of website visited, no interaction between Twitter user was further taken into consideration.

These sites are therefore surfed in our simulation environment. This consist of a virtual box where a browser instance visits a url and record both in page categories and third-party requests. In this scenario we pretend that a user is going through their reading list of sites and by looking at third-party http requests we measure how the advertising changes in the page and adapts to their profile. The user is simulated by a software agent opening the urls and scrolling through the page for a certain arbitrary amount of time.

It is important to note that in our simulated environment the users are not logged a third-party account, like Google, Facebook or Twitter. When the website is accessed a text version of the page is recorded and analysed by our software agent. In page keywords and meta information are extracted and evaluated. We extracted keywords from the actual text of the page by using the Rapid Automatic Keyword Extraction (RAKE) [35] algorithm. Each keyword was then evaluated against Open Directory Project (DMOZ) [31] for classification within top levels categories.

Once the user profile was calculated the advertising profile is evaluated. The advertising profile is extracted from url parameters contained in third-party requests. We have collected information regarding each third-party requests made from each page visits. These parameters are again evaluated against DMOZ for classification

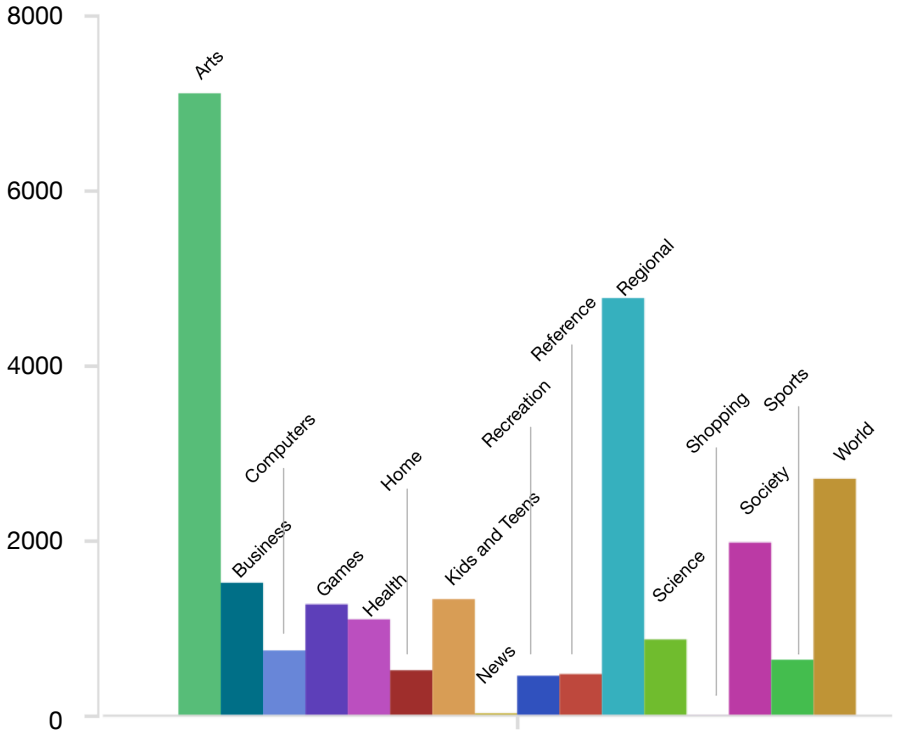

Figure 4.: Here we show an example of user profile expressed in absolute terms by counting the number of keywords in each category for a browsing session. We model user and advertising profiles as histograms of tags keywords a set of predefined categories of interest.

within top levels categories. Please note that we have excluded request made to JavaScript libraries, images and Cascade Style Sheet (CSS) files. We have also excluded same domain requests, since we were only interested in third-parties http calls.

By profiling users' browsing events using a hypermedia document structure we were able to show how each event contains a set of features regarding the user identity and the page that was visited. We have therefore categorised each event by using the keywords contained in the meta information present in the page and the page text itself (Figure: 4). At each event we ere able to calculate an event profile, by measuring the set of keywords introduced by each action performed by the user (i.e. visited a page).

Table 1.: Statistics regarding collected users data

Statistics about collected data			
Categories	16	Users	50
Pages per users	100	Total pages	5000

We profiled 50 users and each user visited a series of 100 pages. In total we analysed 5000 different pages (Table: 1). For each user we calculated how each page contributed to the user profile and also how third-party services adapted to the user profile by returning certain information in form of ads. Information that advertising services request from the visited page may vary in length and type (Listings: 1, 2, 3). Some trackers might include only the referrer *url* and some devices information and user triggered parameters (Listings: 1, 3) while other services might be more lengthily in what is sent from the page (Listing: 2). Some of the information sent through third-party request cannot be categorised, since they include hashed users? ids and internal keywords and code belonging to the tracking service. Other information, like the keywords retrieved from the page (Listing: 2) can be categorised into category of interest that we assume the tracking service uses to profile the user through their interests.

It is interesting to not how among the parameters sent to the third-party tracking services are not included just in page keywords regarding the topic of the page, but also specific browser information. Some of the device's preferences are included in the http headers, like the user-agent identifying the browser and the Operative System. Other information regard how long the page took to load or how soon the content was ready (Listing: 2).

Listing 1: A third-party request to Amazon Ads Service from the nytimes.com homepage. In this example keywords are sent directly as parameters in the http request.

```
Host: aax.amazon-adsystem.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:48.0)
Gecko/20100101 Firefox/48.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://www.nytimes.com/2016/08/29/us/politics/donald-trump-congress-gop-voters.html?hp
Params:
        action: click
        pgtype: Homepage
        clickSource: story-heading
        module: first-column-region
        region: top-news
        WT.nav: top-news
        _r: 0
Cookie: ad-id=A8rOwZ2wOUK4gka1zjqyWNo; ad-privacy=0
Connection: keep-alive
```

Listing 2: A third-party request to krxd.net from a nytimes.com article. This request send different information regarding the article and the browser preferences through http parameters. In addition to the keywords associated with the page, we can see how the request includes information regarding how long it took for the content to be ready $param : t_content_ready$ as well as how much it took for the browser window to load $param : t_window_load$.

```
GET /pixel.gif?
Params:
        source: smarttag
        _kcp_s: nytimes
        _kcp_sc: us
        _kcp_ssc: politics
        _kcp_d: www.nytimes.com
        _kpref_: http://www.nytimes.com/
        _kua_kx_lang: en-us
        _kua_kx_tech_browser_language: en-us
        _kpa_page_type=article
        _kpa_cg: us
        _kpa_scg: politics
        _kpa_pst: News
        _kpa_des: Presidential Election of 2016
        _kpa_per: Lujan Ben Ray
        _kpa_org: Republican Party
        _kpa_author: Alexander Burns and Jonathan Martin
        _kpa_keywords2: Presidential Election of 2016 Elections House of Representatives Politics Action
    Committees Elections Senate Republican Party Lujan Ben Ray Issa Darrell Trump Donald
        t_content_ready: 1792
        t_window_load: 12720
        ...
```

```
Host: beacon.krxd.net
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:48.0) Gecko/20100101 Firefox
    /48.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://www.nytimes.com/2016/08/29/us/politics/donald-trump-congress-gop-voters.
    html?hp&action=click&pgtype=Homepage&clickSource=story-heading&module=first-column-
    region&region=top-news&WT.nav=top-news&_r=0
Cookie: ServedBy=beacon-a262-dub; _kuid_=DNT
Connection: keep-alive
```

Listing 3: A third-party request to facebook.com from a indipendent.co.uk article.

```
Host: graph.facebook.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:48.0) Gecko/20100101 Firefox
    /48.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
DNT: 1
Referer: http://www.independent.co.uk/news/uk/politics/europe-could-go-down-the-drain-
    after-brexit-a7213976.html
Cookie: datr=TbHdVa-yyYq_3UHH_xYR6NGb; fr=0MuKlsg7QM3etJaWt.AWVJMdGky_V9X82TY03Y-wBtGqE.
    BV3bFx.XF.FfD.0.0.BXw9oU.AWVWPpAJ; lu=TggRyE6qvvdCystV9I2G-b0w; _ga=GA1
    .2.1182524233.1441193978; sb=il4HV4ufa1WguCxPntCQagP0; c_user=100007394807876; xs
    =192%3AnWrYMasjLyLusw%3A2%3A1365011662%3A5189; csm=2; s=Aa4hJAWUyE0HSY_M.BXj1Ln; pl=
    n; p=-2; act=1472453154239%2F0; presence=
    EDvF3EtimeF1472453144EuserFA21B07394807876A2EstateFDutF1472453144216Et2F_5b_5d
    Elm2FnullEuct2F1472410836BEtrFA2
    loadA2EtwF240195646EatF1472453143045CEchFDp_5f1B07394807876F2CC
Connection: keep-alive
```

Once we were able to collect and profile readable keywords from http requests, we wanted to know how each page contribute to *how much tracking services know* about our genuine user profile by observing a series of web pages visited. For each users we calculated the TV, the GV_2, the ∞-*norm* and the *KL-divergence* between the partial and the genuine user profile (Figure: 9). The metrics were calculated for 80 visited, while the genuine user profile was calculated over a series of 100 visits. Therefore, in our scenario, if a tracker is present in each visited page they would *know*, in the worst case scenario 80% of the visited pages. Note that the TV gives a measure of the average discrepancy between the probability distributions, while the ∞-*norm* gives the worst case scenario. From our results we see that the worst case scenario and the average one behave similarly.

We have then analysed the case of a tracker that is not present in each of the visited pages. We considered the facebook third-party requests to their services for this experiment. For each user we calculated the TV, the GV_2, the ∞-*norm* and the *KL-divergence* between the partial and the genuine user profile (Figure: 10) for pages where the tracker is present.

Finally we profiled keywords in third-party http requests to Facebook. We wanted to know what information were sent to Facebook for each page visited where the tracker was present. This is important to understand what trackers are able to capture about users preferences if they are not able to *follow* the user across all the pages visited. We assumed that if a tracker is not present on a page, they have no knowledge the user visited it, therefore the partial profile as it is known to the tracker is not modified.

Note also that although none of the users considered in our experiment where logged into Facebook, web pages consistently send data to their third-party tracking services. This means that users are profiled by Facebook even if these are not logged in their platform, and individuals that have decided to opt out of Facebook continue to be targeted and known to their services. This is evident by the request shown on listings 3. A number of browser and device specific information is collected by the http call although the user isn't connected to Facebook. For each users we calculated the TV, the GV_2, the ∞-*norm* and the *KL-divergence* between the advertising profile q and the genuine user profile p (Figure: 11) for pages where the tracker is present. We considered a shorter series of pages (15 pages) following the

intuition that advertising networks might try to form a profile of the user instantly given a small number of visits to similar pages. This was consistent with previous results obtained [2].

We have also analysed network structure among the discovered trackers. By using our footprint model we also considered how tracker domains are linked to pages. In this case we calculated the average degree of the neighbourhood of each node, for nodes corresponding to advertising services. Our results show how it is possible to identify known tracker domains by measuring the average degree of the neighbourhood (Table: 2).

Considering the average degree of the neighbourhood of each node, we can also find out about some interesting properties of the network. We started considering the *in-degree distribution* of the network (Figure: 5). The degree distribution $P(k)$ of a network is defined as the fraction of nodes in the network with degree k. It is particularly interesting to note that the network *in-degree distribution* approximately follow a power law.

Another interesting property to consider is *assortativity*. Assortativity considers the conditional probability that a node of degree k is connected to a node with degree k'. If the probability function is increasing, the network is said to be assortative, showing that nodes of high degree are more likely to connect to nodes of high degree. If the function is decreasing the network is dissortative, meaning nodes of high degree are more likely to connect to nodes of lower degree. We found that the scalar assortative coefficient for the resulting graph is of -0.19 a value that is often found for internet systems [36].

Tracker domain	avg $k_{nn,i}$
tacoda.at.atwola.com	180.0
bcp.crwdcntrl.net	180.0
match.prod.bidr.io	180.0
glitter.services.disqus.com	180.0
ad.afy11.net	180.0
idsync.rlcdn.com	180.0
mpp.vindicosuite.com	180.0
aka-cdn-ns.adtechus.com	180.0
clients6.google.com	180.0
i.simpli.fi	180.0
ads.p161.net'	180.0
dis.criteo.com	180.0
ads.stickyadstv.com	180.0
cms.quantserve.com	180.0
ads.yahoo.com	129.0
graph.facebook.com	118.0
ib.adnxs.com	110.0
rs.gwallet.com	108.0
bid.g.doubleclick.net	98.333
googleads4.g.doubleclick.net	98.333

Table 2.: The table shows the top 20 identified tracker domains based on the average degree of the neighbourhood.

We also generated a partition of SBM and nested SBM of the resulting graph (Figures: 6 and 7). Here we identified how the resulting network partitions and communities corresponds to know tracker domains. This means tracking service

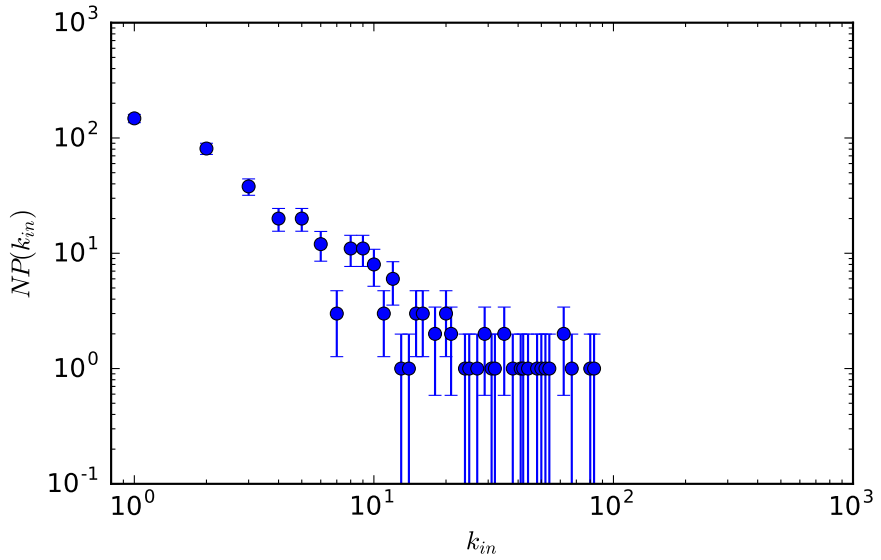

Figure 5.: Degree distribution for the network resulting from the footprint model of users activity. We can see how the degree distribution follows a power law.

exhibit similar properties across the same domain and its structure can be identified through statistical inference over the graph.

Furthermore we also computed page-rank algorithm among the network and identified most connected tracked domains (Figure: 8). Again we were able to spot known tracker domains.

6. Conclusions and future work

We introduced a set of metrics to show how information is sent to third-party tracking services when users surf the web. Because we considered users that were not logged into any identity account, such as Twitter, Google+ or Facebook, we show how third-party service were still able to collect valuable information. We computed the set of metrics for the partial user profile at each page visited. This shows how each page contribute to the actual user profile at the end of a series of websites visited. This means that an advertising network that is present on most of the pages visited possess a large amount of information regarding users and population of users. This information finally allows networks to predict fairly quickly user's preferences and behaviour. We also computed a set of network analysis on our graph model of the user online footprint. We were able to identify known trackers and isolate communities of similar trackers. This aspect is particularly interesting for the development of Privacy Enhancing Technologies for the web. Up to now, anti-tracking technologies have been built to simply stop third-party requests, alternative strategies might instead consider to send bogus information to certain over-connected tracker domains to masquerade the user real profile. At the same time a measurement of the average degree of the neighbourhood of a certain third-party domain can be used to evaluate how *dangerous* this can be considered for the user's privacy. In future research we would like to further explore the graph model introduced, while continuing to understand how quickly web advertising is able to match the served ads with the actual user profile. Particularly we would like to consider real browsing patterns or interactions on social networks possibly

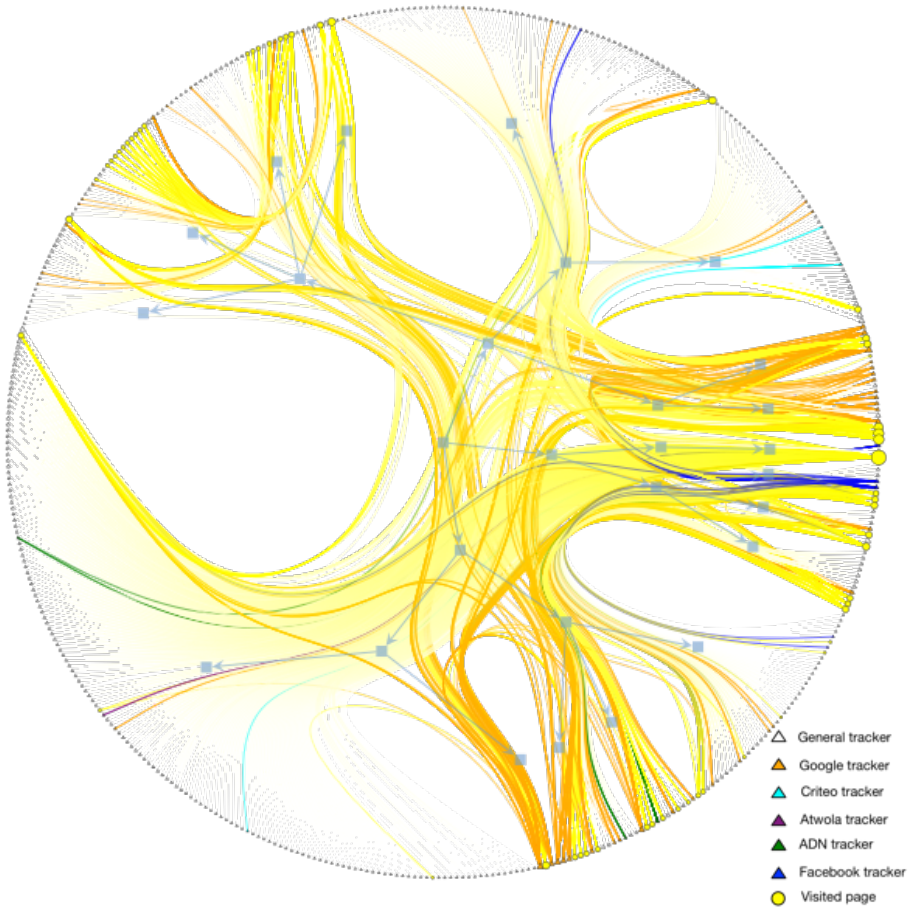

Figure 6.: Block-model decomposition of the network. We can see how we can identify known tracker networks, and how trackers can be grouped into communities that exhibits similar network structure. The blue squares represent the block partition of the network. While the legend is referred to the original nodes, here represented by the shapes on the border.

using a voluntary set of real users. This would allow us to understand if different profiles for the same users can be somehow linked together within similar advertising networks. Moreover, we want to enlarge the set of users analysed by testing on logs from a real world small computer network, while also introducing new metrics to our study. We also believe in the importance to provide users with simple visualisation tools able to show the user their online footprint and allowing them to take action to masquerade their interests profile or simply block certain networks.

Acknowledgements

This work was supported by the Spanish Ministry of Economy and Competitiveness (MINECO) through the "Anonymized Demographic Surveys (ADS)" project, ref. TIN2014-58259-JIN, under the funding program "Proyectos de I+D+i para Jvenes Investigadores", and through the project "INRISCO", ref. TEC2014-54335-C4-1-R, as well as by the Government of Catalonia, under grant 2014 SGR 1504.

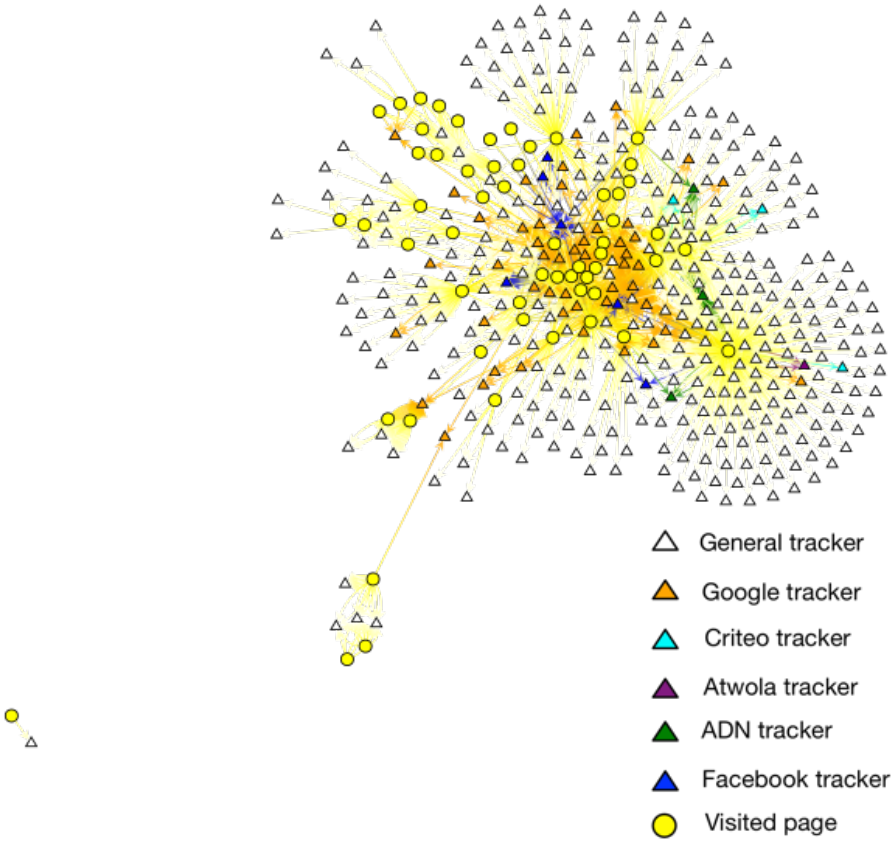

Figure 7.: Blockstate representation of the network of tracking service resulting from our simulation. Here we highlight connections between known tracker networks and visited page.

References

[1] S. Puglisi, D. Rebollo-Monedero, and J. Forné, *On Web User Tracking: How Third-Party Http Requests Track Users' Browsing Patterns for Personalised Advertising*, in *2016 Mediterranean Ad Hoc Networking Workshop, Med-Hoc-Net 2016*, 2016.

[2] S. Puglisi, D. Rebollo-Monedero, and J. Forné, *You Never Surf Alone. Ubiquitous Tracking of Users? Browsing Habits*, in *International Workshop on Data Privacy Management*, 2015, pp. 273–280.

[3] K. Michael and R. Clarke, *Location and tracking of mobile devices: Überveillance stalks the streets*, Computer Law & Security Review 29 (2013), pp. 216–228.

[4] M. Veeningen, A. Piepoli, and N. Zannone, *Are on-line personae really unlinkable?*, in *Data Privacy Management and Autonomous Spontaneous Security*, Springer, 2014, pp. 369–379.

[5] L. Getoor and A. Machanavajjhala, *Entity resolution: theory, practice & open challenges*, Proceedings of the VLDB Endowment 5 (2012), pp. 2018–2019.

[6] P. Eckersley, *Panopticlick* (2011).

[7] K. Boda, Á.M. Földes, G.G. Gulyás, and S. Imre, *User tracking on the web via cross-browser fingerprinting*, in *Information Security Technology for Applications*, Springer, 2012, pp. 31–46.

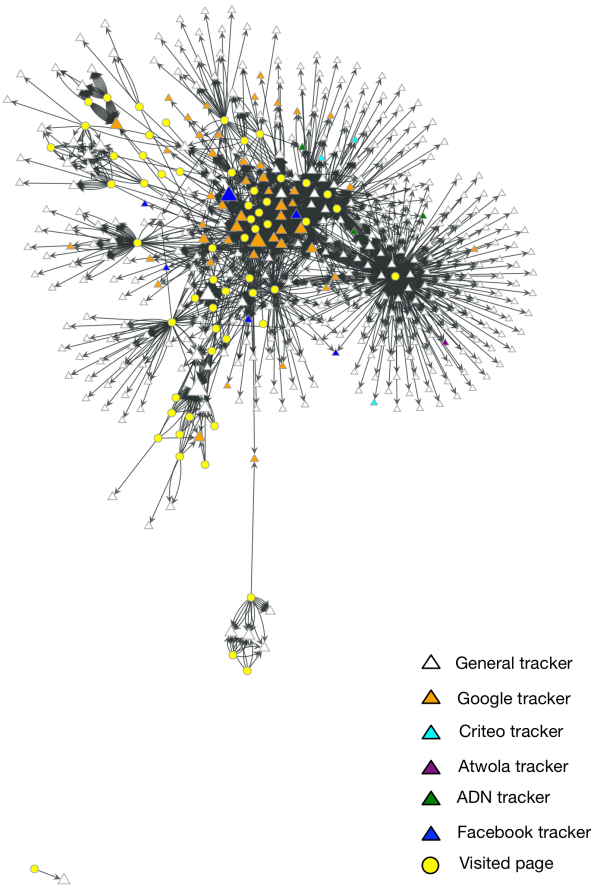

Figure 8.: Pagerank computed over the tracking network. Known tracker domains that are *more connected* can be seen with a bigger node symbol compared to less connected ones.

[8] K. Mowery, D. Bogenreif, S. Yilek, and H. Shacham, *Fingerprinting information in javascript implementations*, Proceedings of W2SP 2 (2011).

[9] C. Castelluccia, *Behavioural tracking on the internet: a technical perspective*, in *European Data Protection: In Good Health?*, Springer, 2012, pp. 21–33.

[10] A. Rao, F. Schaub, and N. Sadeh, *What do they know about me? contents and concerns of online behavioral profiles*, (2015).

[11] V. Kalavri, J. Blackburn, M. Varvello, and K. Papagiannaki, *Like a pack of wolves: Community structure of web trackers*, in *International Conference on Passive and Active Network Measurement*, 2016, pp. 42–54.

[12] S. Schelter and J. Kunegis, *Tracking the Trackers: A Large-Scale Analysis of Embedded Web Trackers*, in *Tenth International AAAI Conference on Web and Social Media*, 2016.

[13] R. Gomer, E. Mendes Rodrigues, N. Milic-Frayling, and M. Schraefel, *Network Analysis of Third Party Tracking: User Exposure to Tracking Cookies through Search*, in *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, Vol. 1, 2013, pp. 549–556.

[14] A. Barrat, M. Barthelemy, R. Pastor-Satorras, and A. Vespignani, *The architecture of complex weighted networks*, Proceedings of the National Academy of Sciences of the United States of America 101 (2004), pp. 3747–3752.

[15] R. Pastor-Satorras, A. Vázquez, and A. Vespignani, *Dynamical and correlation*

8

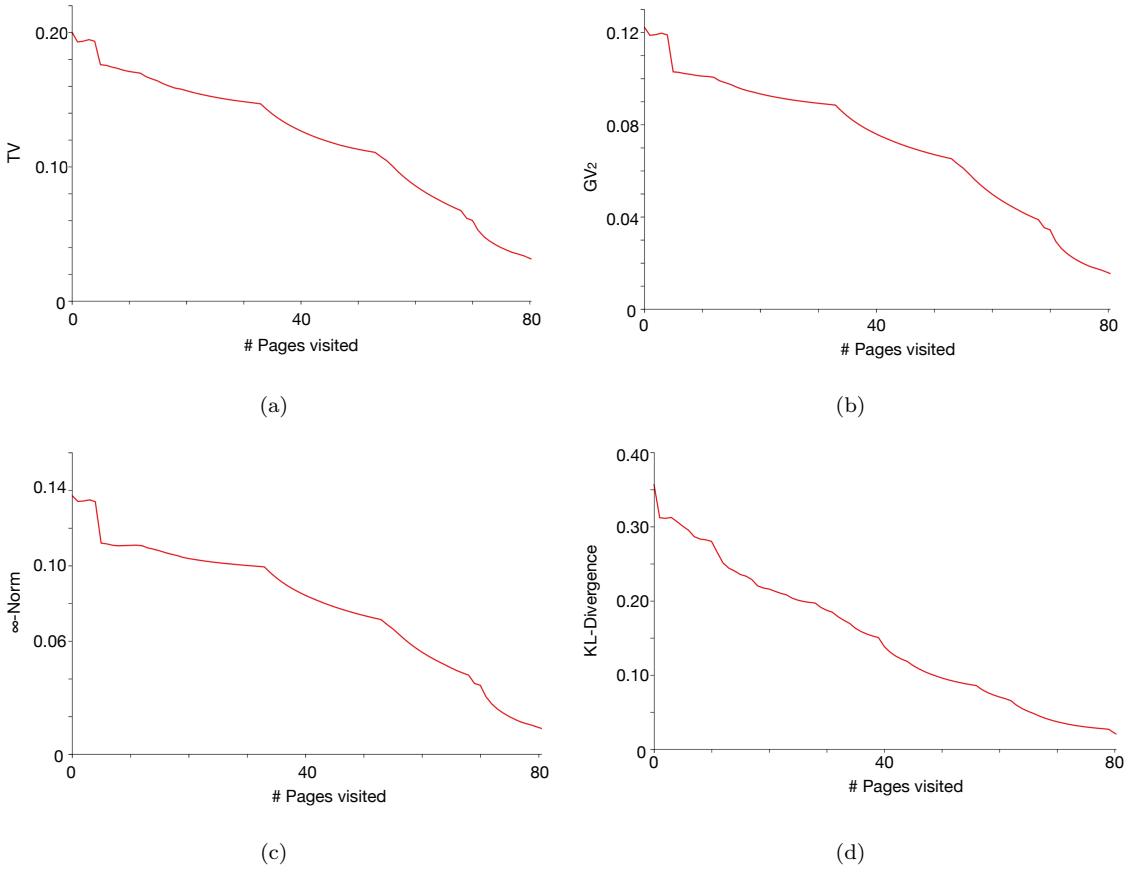

Figure 9.: The figures show how each page visited contribute to the actual user profile. Please recall that we calculated the user profile at the end of the series of 100 web pages visited and we calculated the metrics for 80 visits, giving a 80% estimation. We therefore computed the TV(a), the GV_2(b), the ∞-*norm*(c) and the *KL-divergence*(d) for all pages and averaged among all users.

properties of the internet, Physical review letters 87 (2001), p. 258701.

[16] S. Maslov and K. Sneppen, *Specificity and stability in topology of protein networks*, Science 296 (2002), pp. 910–913.

[17] M.E. Newman, *Assortative mixing in networks*, Physical review letters 89 (2002), p. 208701.

[18] P.W. Holland, K.B. Laskey, and S. Leinhardt, *Stochastic blockmodels: First steps*, Social networks 5 (1983), pp. 109–137.

[19] K. Faust and S. Wasserman, *Blockmodels: Interpretation and evaluation*, Social networks 14 (1992), pp. 5–61.

[20] T.P. Peixoto, *Entropy of stochastic blockmodel ensembles*, Physical Review E 85 (2012), p. 056122.

[21] T.P. Peixoto, *Hierarchical block structures and high-resolution model selection in large networks*, Physical Review X 4 (2014), p. 011047.

[22] T.P. Peixoto, *Efficient monte carlo and greedy heuristic for the inference of stochastic block models*, Physical Review E 89 (2014), p. 012804.

[23] *Privacy badger - https://www.eff.org/privacybadger* .

[24] *Mozilla lightbeam - https://www.mozilla.org/en-us/lightbeam/* .

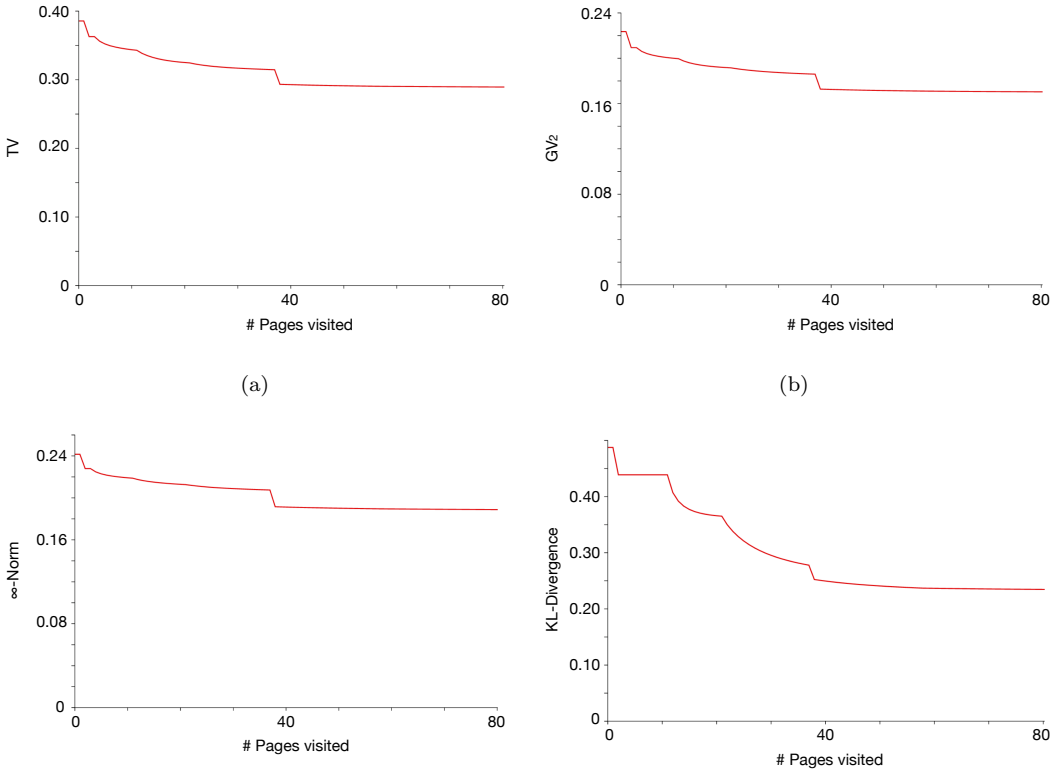

Figure 10.: The figure show the relation between the profile captured by third-party requests to Facebook services and the actual user profile. Please recall that we calculated the user profile at the end of the series of 100 web pages visited and we calculated the metrics for 80 visits, giving a 80% estimation.We therefore computed the TV(a), the GV_2(b), the ∞-*norm*(c) and the *KL-divergence*(d) for all pages and averaged among all users.

[25] *Ghostery - https://www.ghostery.com/* .
[26] *Adblock - https://adblockplus.org/* .
[27] M. Degeling and T. Herrmann, *Your interests according to google-a profile-centered analysis for obfuscation of online tracking profiles*, arXiv preprint arXiv:1601.06371 (2016).
[28] B. Ard, *Confidentiality and the problem of third parties: Protecting reader privacy in the age of intermediaries*, (2013).
[29] J. Parra-Arnau, D. Rebollo-Monedero, J. Forné, J.L. Muñoz, and O. Esparza, *Optimal tag suppression for privacy protection in the semantic Web*, Data, Knowl. Eng. 81–82 (2012), pp. 46–66, URL `http://dx.doi.org/10.1016/j.datak.2012.07.004`.
[30] J. Parra-Arnau, A. Perego, E. Ferrari, J. Forné, and D. Rebollo-Monedero, *Privacy-preserving enhanced collaborative tagging*, IEEE Trans. Knowl. Data Eng. 26 (2014), pp. 180–193, URL `http://dx.doi.org/10.1109/TKDE.2012.248`.
[31] *Open directory project - http://www.dmoz.com* .
[32] J. Parra-Arnau, D. Rebollo-Monedero, and J. Forné, *Measuring the privacy of user profiles in personalized information systems*, Future Generation Com-

10

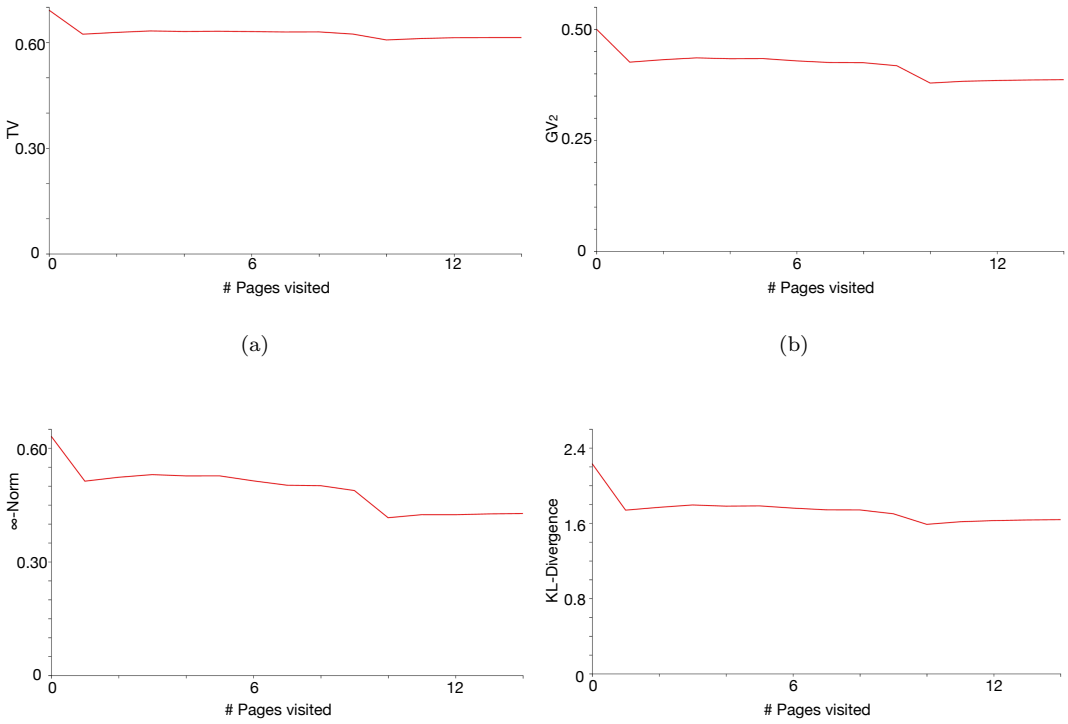

Figure 11.: The figure show the relation between the profile sent by third-party requests (q_n with $n \in [1, N]$) to Facebook services and the actual user profile. Please recall that we calculated the user profile at the end of the series of 15 web pages visited. We therefore computed the TV(a), the GV_2(b), the ∞-$norm$(c) and the KL-$divergence$(d) for all http calls and averaged among all users.

puter Systems 33 (2014), pp. 53–63.

[33] D. Rebollo-Monedero, J. Parra-Arnau, and J. Forné, *An information-theoretic privacy criterion for query forgery in information retrieval*, in *International Conference on Security Technology*, 2011, pp. 146–154.

[34] T.P. Peixoto, *Parsimonious module inference in large networks*, Physical review letters 110 (2013), p. 148701.

[35] S. Rose, D. Engel, N. Cramer, and W. Cowley, *Automatic keyword extraction from individual documents*, Text Mining (2010), pp. 1–20.

[36] R. Noldus and P. Van Mieghem, *Assortativity in complex networks*, Journal of Complex Networks (2015), p. cnv005.