Enhancing user awareness and control of web tracking with ManTra

Davide Lo Re
University of Rome 'La Sapienza'
Email: lore@di.uniroma1.it

Claudio Carpineto
Fondazione Ugo Bordoni, Rome
Email: carpinet@fub.it

***Abstract*—Web trackers can build accurate topical user profiles (e.g., in terms of habits and personal characteristics) by monitoring a user's browsing activities across websites. This process, known as behavioral targeting, has a number of practical benefits but it also raises privacy concerns. Most existing techniques either try to block web tracking altogether or aim to endow it with privacy preserving mechanisms, but they are system-centered rather than user-centered. Nowadays, the majority of users want to have some degree of control over their privacy, while their perspectives and feelings towards web tracking may be different, ranging from a desire to avoid being profiled at all to a willingness to trade personal information for better services. Regardless of a specific user's preference, from a technical point of view there is is no simple way for him/her to monitor, let alone to influence, the behavior of web trackers. In this paper, we describe an approach which makes users aware of their likely tracking profile and gives them the possibility to bias the profile towards both ends of the web tracking spectrum, either by improving its accuracy beyond the tracker capabilities (thus emphasizing behavioral targeting) or by filling in false interests (thus increasing privacy). This goal is achieved by simulating the process of learning a user profile on the part of the tracker and then by retrofitting a web traffic suitable for producing the desired profile. Our approach has been implemented as a web browser extension called ManTra (Management of Tracking). The system has been evaluated in several dimensions, including its ability to learn an accurate ad-oriented user profile and to influence the behaviour of a commercial tool for web tracking personalization; i.e., Google's Ads Settings.**

I. Introduction

The browsing behaviour of web users is being constantly monitored by a number of ad networks making use of increasingly sophisticated tracking techniques. The collected data associated with the visited pages are used to infer user interests across multiple sites and build updated user profiles, usually in the form of topic trees. User interest profiles enable behavioural targeting (aka interest-based advertising), beyond simple context-aware advertising: when someone's profile relates to an ad category, the advertiser can target that user even if they are browsing on unrelated sites. At the same time, behavioural targeting raises privacy concerns because the visited pages can reveal a great deal about a user's habits and characteristics; e.g., location, purchases, employment and financial status, sexual and political orientation, medical conditions.

Preventing web tracking by removing cookies and disabling non-cookie, privacy-related technologies has proved to be very difficult in practice and it is likely to affect the user's web experience. More recently, the research focus has shifted from web tracking blocking to enriching web browsing with privacy preserving mechanisms; e.g., [1], [2], [3], [4]. However, while most proposed systems are effective at providing privacy-enhanced web browsing in a lab setting, their large-scale deployment may be more problematic because it is hindered by technical, organisational, and economic issues, as discussed in Section II. Furthermore, one common feature of these approaches is that they are system-centered rather than user-centered, because they do not take into account the needs and preferences of single users.

Most recent studies suggest that a large majority of users are concerned with their lack of web privacy control [5]. In the past years, a considerable effort has been made to provide users with mechanisms to control the presence and the activity of web trackers. These systems work at the level of data collection and aggregation. Well known examples are Fourth party [6] and Tracking the Trackers [7]. However, users are not allowed to know what web trackers have inferred about them based on these data. The inference level is more important for users. For instance, [8] advocates greater transparency about inferences enabled by data aggregation rather than about data aggregation itself, because users are not typically aware of the consequences of aggregation.

Not only are users unaware of their tracking profiles, but they cannot express their preferences. It is well known that not all users are equally concerned with their web browsing privacy. According to a recent study [9], 46% of Internet users say it is very important that only they or those they authorize have access to the websites they browse, 23% say that it is somewhat important, while a robust minority of 28% say that it is not particularly important. User preferences have been investigated in a few research works suggesting that individuals assign markedly different values to the privacy of their data, to a much larger extent than observed for other private goods [10]. It has also been pointed out that users are willing to give up their privacy if they perceive that they get something of value in exchange for their data [11], such as obtaining personalized services and more relevant ads. The anticipated variability of user preferences can be tackled by favouring a more flexible approach to web tracking management that makes policy alternatives simpler and clearer and lets a user express their preferences.

A remarkable attempt to give a user more knowledge

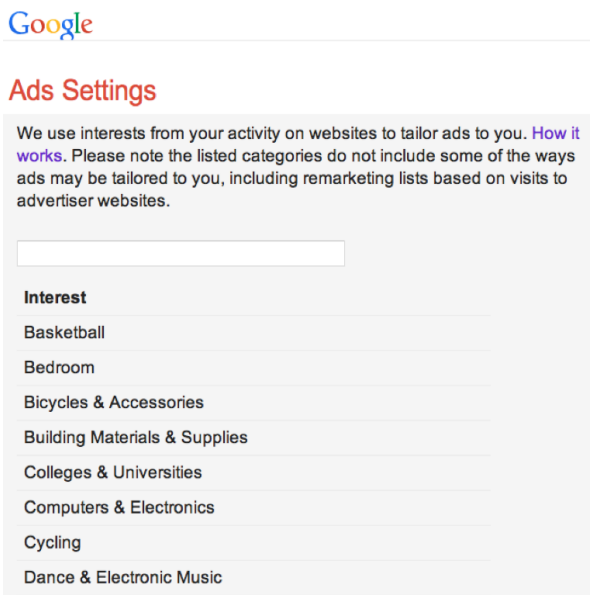


Fig. 1. Portion of Google Ads list of interest categories associated with the second author of this paper from his browsing activity.

and control over their tracking profile is represented by the decision made by some ad networks – most notably Google – to publish their data, probably induced by the pressure of regulators and media. As an illustration, we show in Figure 1 the categories associated with the second author of this paper in the 'Interest' section of his Google's Ads Settings profile. The topics inferred by Google are usually very specific and accurate.[1]

However, Ads Settings is one of the few systems of this kind that makes these data available to the user for display and editing.[2] Furthermore, there is no guarantee that this service will not be discontinued at some time. The profiles created by other ad networks are simply not accessible and therefore users have no control at all over what is going on on their side. The first goal of our research is to provide a user with the possibility of knowing what ad networks have most likely inferred about him/her, without relying on the willingness of a specific ad network to do so using its proprietary data and inference capabilities.

Also, the process of editing one's own tracking profile made available by an ad network may be long and tedious because a user has to manually select categories of interest from a very large taxonomy. If there are more published profiles, a user has to update all of them using different interfaces and taxonomies. Furthermore, these operations should be repeated over time, as interests (and especially advertising-related interests) change over time. Our second goal is to let a user express their preferences at an abstract level (e.g., privacy, neutral, targeting) and have the system build a suitable target tracking profile reflecting such preferences.

The next issue we address is how to scale this approach to the whole tracking ecosystem. The universe of ad networks is far larger than the few who were willing to publish their data. The others will continue to operate without control and without the guarantee that they are unable to build an accurate tracking user profile when a user wants to protect privacy. Even the companies with a manually editable profile from which we opted out will in principle be able to keep inferring our true interests from the web browsing data. Our third goal is to simultaneously influence the inferences made by all ad networks (not only those who published their data) according to user preferences.

To achieve these goals, we present a comprehensive method consisting of several steps assembled in a pipeline. We first mimic the learning process behind the construction of user tracking profiles by ad networks, by constructing an ad-oriented classifier and learning a weighted topic tree for a user from the browsing history. This tree can be thought of as the tracking profile learned by an ad network that has access to the user's browsing traffic. Then we acquire the user's high-level preferences and automatically transform them into a target tracking profile, as opposed to manually entering a number of detailed categories of interest and editing them over time (as with Google Ads). Finally, we move the true user profile towards the target profile, by retrofitting suitable synthetic web traffic to the target profile and by monitoring the convergence of the true profile to the target profile through an ad-hoc similarity measure. This approach is implemented in a system called ManTra (which stands for Manegement of Tracking), available as a Firefox addon. The two most important features of ManTra – the ability to learn ad-oriented user profiles and to influence the construction of tracking user profiles by ad networks – are evaluated.

The main contributions of this paper are the following.

- Users can see their likely tracking profiles, represented as ad-oriented weighted trees learned from their web history. As a byproduct, we created a specialised topic taxonomy with associated web pages that is made available for reuse. This is the first dataset of this kind (to our knowledge).
- User preferences are made explicit, whereas this dimension has been somewhat underexploited in the development of web privacy management systems. Our framework can be tailored to the specific goals of a user, from militants to compromisers of web privacy.
- A single framework capable of biasing user profiling towards either end of the web tracking spectrum; i.e., privacy or targeting.

The remainder of the paper has the following structure. In Section II we review related works on web tracking. In Section III we describe the architecture and the single components of ManTra. Section IV is dedicated to the experiments, including a user study, and Section V offers some conclusions

[1]To see how Google has profiled you for targeting (and to refine or opt-out), check out Ads Settings - Google, at https://www.google.ca/settings/ads.

[2]Two other networks of which we are aware are BlueKai and Yahoo!, but as at the time of writing this paper they essentially provide a user with opt-out tools, without displaying their tracking profiles. A similar restricted opt-out approach is taken by the recent Network Advertising Inititiative and Digital Advertising Alliance.

II. Related work

A. Hinder web tracking

A number of techniques are available for making web tracking more difficult, or even impossible in some cases, by manipulating browser state. Blocking cookies and certain invisible page elements (e.g., scripts, pixels, iframes) altogether or only from domains belonging to known trackers is a simple and popular strategy, exemplified by Ghostery[3]. This approach however may heavily affect the user's web browsing experience and it does not allow behavioural targeting. Furthermore, ad networks may develop effective replacement techniques such as fingerprinting [12] and others, as trackers typically employ a combination of tracking behaviours [13]. The do-not-track header is a more principled way of interfering with tracking because it is a preference that users can set in web browsers to inform websites that they do not want to be tracked. However, it entirely relies on the cooperation of ad networks, whose business model opposes compliance.

B. Privacy-enhanced targeted advertising

This research line is an attempt to allow behavioural targeting while at the same time protecting user privacy. The general idea is to move interest mining to user endpoint and to create a middle layer between users and advertisers (e.g., in the form of proxies, auction platforms, dedicated hardware) to manage the flow of information privately and securely. A few examples are [1], [2], [3], [4]. These techniques are effective from a theoretical point of view but they have specific requirements and rely on technological, organizational, and economic assumptions that are often difficult to fulfill, such as requiring trusted third parties or a willing involvement of ad networks. As a matter of fact, their deployment has been lacking so far, with very few exceptions [14]. Furthermore, they act as a black box and users are really not able to exert any control over their profiles.

C. Inform the user

A lot of recent research has focused on tracking the trackers [7], in an attempt to reveal the activities of web parts that are not transparent to the average user. Two well known examples are Lightbeam[4] and FourthParty[5] [6], which is probably the most complete tool for detecting tracking mechanisms, including supercookies and active fingerprinting. These systems are very useful for detecting the activities of trackers and increasing the awareness of users, but they work exclusively at the level of data collection and aggregation. They do not analyze what kind of personal information third parties learn about users, e.g., in the form of an interest tree. In addition, the main technical solutions for giving user more control over third-party web tracking are opt-out cookies and blocking, which are a realistic solution only for advanced users.

[3]https://www.ghostery.com

[4]https://www.mozilla.org/en-US/lightbeam/

[5]http://fourthparty.info/

D. Web tracking personalization

Displaying editable user profiles is a well known strategy for web service personalisation; e.g. web search [15] and recommender systems [16], [17]. In the web tracking domain, the most straightforward manner is to manually input a user profile (at least for those ad networks that provide the user with this capability), but this approach is effective only for specific trackers and it may be very tedious in practice. To overcome these limitations, one can try to purposefully modify the data that trackers collect to draw their inferences. One recent system of this kind is Synthoid [18]. It asks the user to select a set of ODP topics of interest and then performs fake visits of URLs associated with those topics in an attempt to imprint tracking profiles; i.e. Ads Settings.[6] Essentially, the system is concerned only with step 4 of ManTra (see Figure 2), where the user manually constructs the input to step 4 by clicking on the ODP categories. In addition, as Synthoid does not maintain its own user profile, it cannot evaluate whether the biased tracking profile is moving in the right direction, let alone convergence to a desired profile.

Unlike most earlier works, Mantra is a truly user-centered system. It combines the advantages of revealing what web trackers likely know about a user and letting the informed user decide if/how to change that knowledge according to his/her preferences. Mantra does not hide the user's online behaviour because this is outside of its scope. On the other hand, as all traffic is generated directly from the user's machine, Mantra can be integrated with any other privacy-by-design advertising technique, thus providing a framework for building a more comprehensive web tracking management system that is both flexible and safe.

III. Description of ManTra

A. ManTra workflow

Mantra consists of four main steps. In step 1 we build an ad-oriented hierarchical classifier by leveraging a specialized ad-oriented topic taxonomy and creating suitable training sets. In step 2 we infer a weighted user profile from the user's browsing history using the classifier from step 1, where the category weights reflect the relevance and recency of visited pages associated with the single categories. In step 3, we build a target tracking profile based on the user's preferences, using noisy categories or relevant categories learned from the user's bookmarks depending on whether the user has chosen the privacy or targeting option. In step 4, we use suitable synthetic web traffic to move the true profile to the target profile and monitor convergence through an ad-hoc similarity measure between weighted trees. The web pages in the synthetic traffic are associated with noisy or relevant categories, depending on the user's preferences. The overall workflow of ManTra is illustrated in Figure 2 and its components are described in detail in the following sections.

[6]Ads Settings has not been investigated only by Synthoid. It attracted other experimental work mostly focusing on the interactions between visiting webpages and the corresponding changes in Ads Settings profile and the ads shown by Google [19], [20].

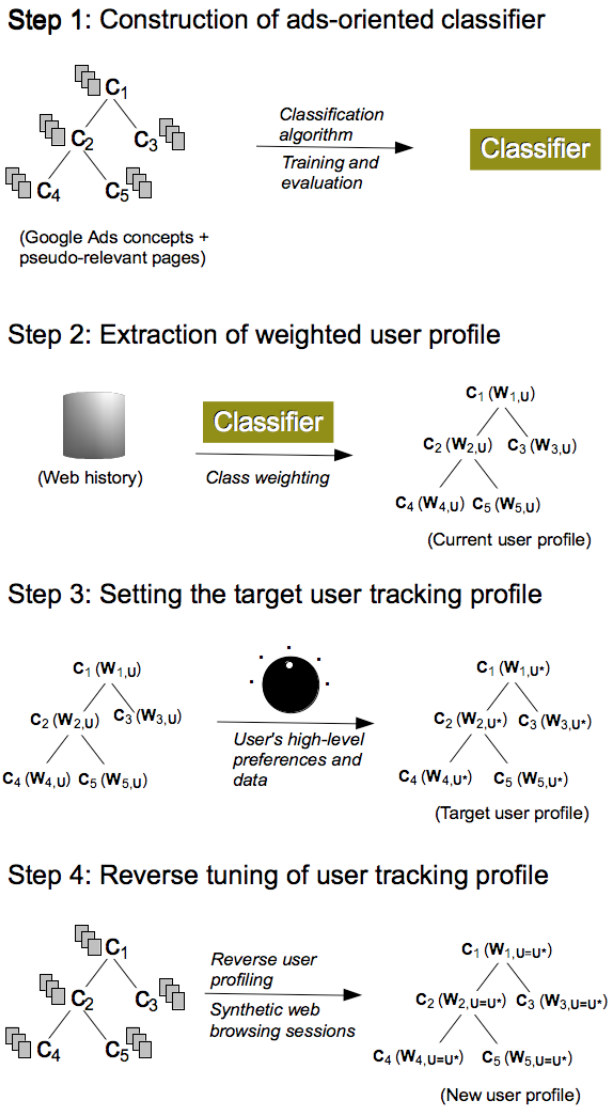


Fig. 2. Workflow of ManTra.

B. Ad-oriented classifier

In order to build a topic tree from a set of visited pages, ManTra must solve a hierarchical multiclass page classification task.[7] Following [21] and [22], ManTra uses several flat multiclass classifiers (one for each node in the hierarchy), where each flat multiclassification is performed through multiple binary classification with a one-vs-all approach [23]. The training data are partitioned into node-specific subsets and the multiclass classifiiers at each node are trained independently.[8]

To perform binary page classification we use a Naive Bayes classifier, which produces a real-valued confidence score for its decision and is known to be effective for text classification problems. We used only URLs and titles, partly because of efficiency reasons and partly because using a richer set of information may have a very limited impact on classification performance [26]. Finally, following [27]), the actual input to the binary classifier consists of character ngram-based features extracted from individual path components and from single words in the title.

The next step is to define a specific topic tree with suitable training sets to train the hierarchical multiclass classifier. One specialized topic tree for ad purposes has been developed by Google's Ads Settings. To our knowledge, this is the only taxonomy of this kind made available for consultation.[9] For each category in the tree, we queried its name together with the name of its ancestors to a general search engine, getting URLs and titles of the first 100 search results. We assume that the first results returned by the search engine are relevant to the category query, and thus can be seen as (positive) instances of that category. This procedure is similar to the technique used in automatic query expansion to find pseudo-relevant documents [28]. We then attribute all the results of a child to all its ancestors, and cut the tree so that it has only 2 levels of 23 and 236 categories. As each category inherits the instances of its descendants, the number of instances per category varies from 100 to a few thousand. The full training dataset containing 83,648 URLs and titles was used to train 259 binary classifiers (e.g., one classifier outputs 'Sports' or 'non-Sports') and will be made publicly available for reuse.

C. Constructing and comparing weighted user profiles

The classifier described in the preceding section can be used to assign any page in the web history to a path in the topic tree, activating the corresponding categories in the user profile. However, this is not enough to allow us build an accurate user profile, because we need to assess the relative importance of topics for any specific user.

Following Ads Settings' explanation that "Our system looks at the types of pages a user visits, taking into account how recently and frequently those pages have been visited, and then associates their browser with relevant interest categories", we assume that the contribution made by a visited page to a given category depends on three main factors, namely the confidence that the page can be assigned to that category, the number of visits, and their recency The classification confidence is returned by the Bayesian classifier, while the recency of visits can be conveniently modeled using the exponential distribution. Using this approach, the weight of category x at time t^* can be expressed as:

$$w_{x,t^*} = w_{\hat{x},t^*} \cdot \frac{\sum_{u \in x} p(x|u) \cdot \frac{1}{\beta} e^{-\frac{\Delta_t}{\beta}}}{\sum_{c \in \{S(x) \cup x\}} \sum_{u \in c} p(c|u) \cdot \frac{1}{\beta} e^{-\frac{\Delta_t}{\beta}}}$$

where $p(x|u)$ is the confidence of the category x given the instance u, Δ_t is the difference between the time of visit and the time t^* of profile creation, β is the survival time

[7] In multiclass classification the instances must be partitioned into $k > 2$ classes. It should not be confused with multilabel classification, where multiple classes are to be predicted for each instance.

[8] Other approaches that try to make a better use of the hierarchical structure have been proposed, usually at the cost of increased complexity; e.g. [24], [25].

[9] See https://support.google.com/ads/answer/2842480

of every page visit, and $\hat{x}$ and $S(x)$ are, respectively, the parent node and the set of siblings of category x. It can be shown (details are suppressed due to space limitations) that this formula ensures the consistency of weights across the tree by redistributing the weight of a node to its children..

Not only is ManTra able to learn a user profile, but it is also equipped with the ability to compare different user profiles. This enables us to measure how a user profile varies over time and assess whether it is shifting towards the desired end of the web tracking spectrum when injecting the synthetic browsing traffic. Measuring the similarity between two weighted trees is, however, not straightforward.[10] We use a vector space model approach, transforming the trees into weighted vectors and then applying conventional weighted vector matching to compute their similarity.

D. Setting the target profile with user preferences

Mantra gives a user the possibility to express three types of preferences: privacy, neutral, targeting. After the user has chosen one mode, the system sets the corresponding target profile. In the 'privacy' mode, an obfuscated target profile is built by randomly choosing paths from the topic tree and randomly assigning consistent weights to the selected categories. In the targeting mode, a user-focused target profile is set by running the classifier not only on the user's history but also on the on the bookmarks, which are valuable and unknown to trackers. In the neutral mode, no operation is performed because it is assumed that a user is happy with their current profile.

E. Automatic tuning of tracking profile

The process of visiting fake URLs starts after the user has entered his/her preferences at time t^* and the system has generated the corresponding target profile (unless the 'neutral' option has been selected). The goal is to make the biased user profile resemble the target profile as closely as possible. We define the objective function of profile tuning at time $t > t^*$ as the maximization of the similarity between the target profile set at time t^* and the current profile at time t. The current profile is built from the union of the natural traffic at time t^* and the synthetic traffic from t^* to t (we assume for the sake of simplicity that the contribution made by the natural traffic after t^* is negligible). Two consecutive fake visits are separated by a fixed time interval and the similarity between profiles is computed as illustrated in Section III-C.

The next question that must be addressed is how to select fake URLs. We adopt a simple iterative greedy strategy, based on selecting one URL at a time in the target tree and re-evaluating the objective function associated with the new current profile until convergence.

F. ManTra's user interface

ManTra interface presents three tabs for the user to select from: the *Know* tab, which builds and visualises the current user profile; the *Decide* tab, which lets the user express his/her preferences about web tracking – 'Privacy', 'Targeting', 'Neutral' – and drives the generation of synthetic traffic; the *Check* tab, which visualises information about the generated traffic as a form of system feedback. As an illustration, in Figure 3 we show a screenshot of ManTra with the *Know* option. The interest tree is presented as a sorted, expandable list of top-level categories, where each category has a bar associated with it which reflects its importance. ManTra is available as a web browser extension for Mozilla Firefox, written in JavaScript. To make the system easily available to the public, we plan to publish it using Mozilla Add-ons directory.

Fig. 3. A screenshot of ManTra's user interface.

IV. Evaluation

We evaluated the effectiveness of the two main features of ManTra, namely the ability to learn ad-oriented user profiles and to influence the tracking profiles displayed by Google's Ads Settings.

A. Classification performance evaluation

1) Automatic evaluation: We first evaluate whether ManTra's classifier is able to correctly classify unseen web pages described by URLs and titles. For this purpose, we performed ten-fold cross validation over the balanced sets of positive and negative samples associated with the categories at the top level, letting the size of n-grams vary from 2 to 6 and measuring the corresponding classification accuracy (in percentage). The results are shown in the first row of Table I. The first finding is that the performance initially improves and then stabilizes as the n-gram size grows. This is consistent with the fact that using very short n-grams increases the chances of matching items of unwanted categories while using long n-grams may result in a failure of partial matching items of the right category. Table I also shows that the average accuracy obtained for a range of n-gram sizes was quite good (on an absolute scale) and comparable to that reported for a similar hierarchical classification task on the DMOZ dataset [27], despite using a relatively small number of automatically-generated training instances.

[10]The techniques developed for unweighted trees such as tree edit distance and alignment distance [29] are not easily applicable to node-weighted trees and are computationally inefficient.

TABLE I
AVERAGE CLASSIFICATION ACCURACY (%) FOR FIRST-LEVEL AND SECOND-LEVEL CATEGORIES AS A FUNCTION OF N-GRAMS SIZE.

	2	3	4	5	6
First-level categories	69	84	85	87	85
Second-level categories	47	65	68	70	69

We also measured the classification performance on the second-level categories. To take advantage of the hierarchical structure (as opposed to e.g. using a flat representation), we first ran the top-level classifier and then used its output to run the corresponding second-level classifier. The results, reported in the second row of Table I, confirm the findings obtained for the first-level categories. The performance figures seem notably good, considering that the subcategories of a given top category resemble one another more closely than the top categories do and are thus more difficult to discriminate between. Overall, these findings suggest that the training instances associated with each category are of good quality and that the hierarchical classifier is effectively able to assign a path of the Ads Settings taxonomy to a web page. In ManTra, we set the size of n-grams to 5.

While this experiment suggests the technical feasibility of our approach to page classification, the question remains as to how representative are the top 100 search results for Ads Settings categories of a user's actual history. This issue is addressed below.

2) User study: We conducted a user study aimed to assess the accuracy of the profiles generated by ManTra, focusing on the top level categories. We tested 15 participants in the experiment. They were researchers or members of the administrative staff affiliated in the same institute as the authors of this paper, with good experience in Internet usage.

For each subject, we preliminarily imported in Firefox (on his own computer) the browsing history from the subject's preferred browser. Then ManTra was downloaded, installed as an extension to Firefox, and ran in the 'Know' mode. After seeing the profile returned by ManTra (the profiles contained on average about ten categories) and becoming acquainted with the full list of 23 top level Ads Settings categories, the subject was finally administered a Google form questionnaire with the three following questions.

Q1. To what extent does the displayed profile contain categories that do not match well your browsing behavior?

Q2. To what extent does the displayed profile not contain categories that would match well your browsing behavior?

Q3. What is the global quality of the displayed profile?

A 10-point Likert-type answer continuum was used; i.e. 1 = very low, 10 = very high. For Q1 and Q2, the lower the score the better the profile; for Q3, the lower the score the worse the profile. Figure 4 shows the number of responses for each Likert level in each question. The Q1 and Q2 responses are characterized by an extremely positive value of the mode (= 1 for both) and by a very positive value of the median (= 3 for Q1 and = 2 for Q2); the better value of the median for Q2 can be explained with the observation that recalling an undetected category is harder than noticing an inappropriate one. The Q3 responses exhibit a very positive value of the mode and median (= 8 for both) and a very low statistical dispersion; i.e., interquartile range = 1.

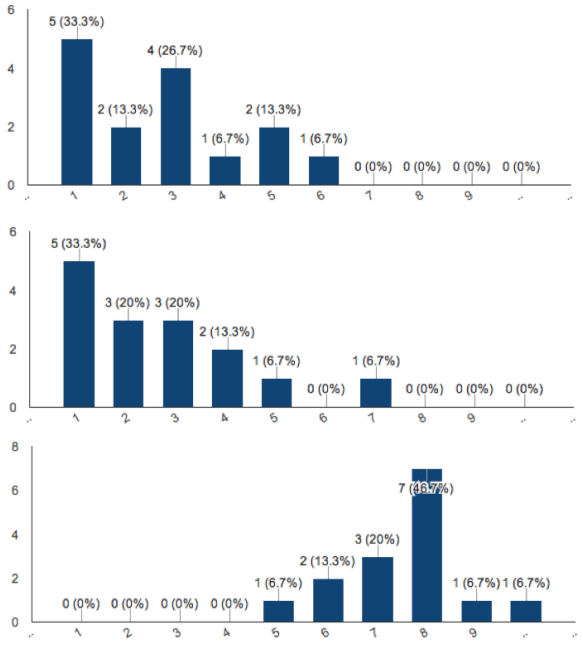


Fig. 4. Analytical summary of the subjects' responses to questions Q1 (top), Q2 (middle), and Q3 (bottom).

Overall, this experiment suggests that the profiles built by ManTra contained very little wrong or missing information and reflected quite well the browsing behavior of the real users tested in the experiment. This kind of evidence is complementary to the previous results about the automatic evaluation of ManTra's classification accuracy.

B. How ManTra affects Ads Settings

1) Experimental setting: The main objective of this set of experiments was to evaluate how fast and how well the synthetic web traffic generated by ManTra is able to influence Ads Settings. We adopted a simplified evaluation scenario, in which initial profiles are empty and the natural web traffic is absent. All our experiments were performed by creating a new instance of the browser with empty cache and cookies, followed by opening a sequence of URLs and monitoring the corresponding changes in the Ads Settings profile. To simulate human web surfing, as opposed to e.g., web crawling, in all our experiments the mean of the interval of time between two consecutive visits (sampled from an exponential distribution) was set to 15 seconds. Another general design feature was that the visit of a web site simply consisted of opening a web connection, without performing any operation on the opened page. We experimentally checked that using more complex

visit strategies, such as randomly clicking on a few links in the downloaded page, was not beneficial because the impact on the tracking profile was both slower (thus suggesting that page navigation is not an essential indicator for validating user interest) and less accurate (due to topic drift).

2) Speed of impact: The number of visits necessary to activate a target category may depend, of course, on the specific policy adopted by a web tracker and is also affected by the given experimental constraints. In our case, for instance, the visit to a web page, no matter how relevant to a given category, will never trigger a profile change if the tracker did not track that page. For each top-level category, we iteratively visited randomly-selected URLs associated with the category until it appeared in Ads Settings profile, and measured the number of visits performed prior to its activation. The results, averaged over 50 runs, are shown in Figure 5 (a). The main finding is that the activation of target categories was fast. Some categories required less that 10 visits; the average number of visits was 14,5 (which take less than 4 minutes in our default setting). In the light of these findings, our assumption that the impact of the natural web traffic is negligible seems justified. Figure 5 (a) also shows that the differences between the single categories were notable. The likely reason is that they have a varying number of associated URLs that can be tracked by trackers, depending on the category content. An example is the category 'science', one of the slowest categories in Figure 5 (a). We found that a large proportion of URLs associated with 'science' consist of academic web sites with no third parties cookies nor commercial ads, which are less likely to be tracked by ad networks.

3) Quality of impact: Now we turn to the experiments aimed to assess how well the induced Ads Settings profile replicates a given target profile. As categories in the Ads Settings profile are not weighted, we used the Jaccard index as a similarity measure between the two profiles: $Sim = \frac{|X \cap Y|}{|X \cup Y|}$, where X and Y are the two profiles to be compared, seen as a set of top- or second-level categories. We used the same runs generated for the experiments described in the preceding section. When the top-level category being tested appeared in the Ads Settings profile, we measured the similarity between the target profile (i.e., the category itself) and the Ads Settings profile. The results, averaged over 50 runs, are shown in Figure 5 (b). All categories but one yielded a profile similarity value greater than 0.5, implying that they were more frequently activated alone (i.e., with zero noise) than with some non-target categories. This seems a good result considering that some URLs can be naturally attributed to multiple categories and that we found evidence that Ads Settings behaves as a multiclass classifier, whereas our URLs have a single category label. If we consider the second-level categories, the similarity value was lower (i.e., it dropped from 0.63 to 0.41), consistent with the fact that it is more difficult to discriminate between categories that are more similar in content. On the other hand, an activated non-target category may still point to a relevant macro-topic of interest (i.e., its parent in the category tree) if it is a sibling of the target category. In our experiments, for instance, the target profile 'Computers+CAD' caused the activation of another seven categories, all of which seem relevant: "Arts & Entertainment/Visual Art & Design", "Business & Industrial", "Business & Industrial/Business Services", "Business & Industrial/Manufacturing", "Computers & Electronics", "Computers & Electronics/Software", "Science".

To evaluate what happens as the size of the target profile grows, we designed another experiment. We randomly chose a target profile consisting of a certain number of categories and visited one randomly selected web site associated with each target category. This cycle was iterated up to a maximum number of visits (i.e, set to 500) until all the target categories had been activated in the Ads Settings profile. When the algorithm stopped, the profile similarity was measured using again the Jaccard index. We found that the profile similarity was, in general, high (i.e. > 0.5) and that it increased as the size of the target profile grew, because it becomes more likely that a category erroneously activated by a URL will coincide with some other target category.

In the experiments described above, the similarity between the target and the commercial tracking profile was evaluated on a per-run basis; i.e., on the moment when all target categories in *each* run were activated in the tracking profile. This gave us a sort of upper bound for average profile similarity. Now we want to study how profile similarity varies as a strict function of the number of visits; in particular, what happens if we keep visiting fake URLs associated with target categories, even after their activation. This may serve to better model a worst case scenario in which we do not have direct access to the profile set by a specific tracker. It also provides an indication of the stability of acquired profiles with respect to redundant synthetic traffic. In general, as we perform more visits, there are more chances that both the target categories as well as other noisy categories will be activated. Figure 5 (c) shows that the profile similarity initially increases and then becomes nearly constant, thus suggesting that the two competing factors balance each other. It is likely that the curves stabilize because the set of URLs associated with each category can, at most, activate a small number of different categories.

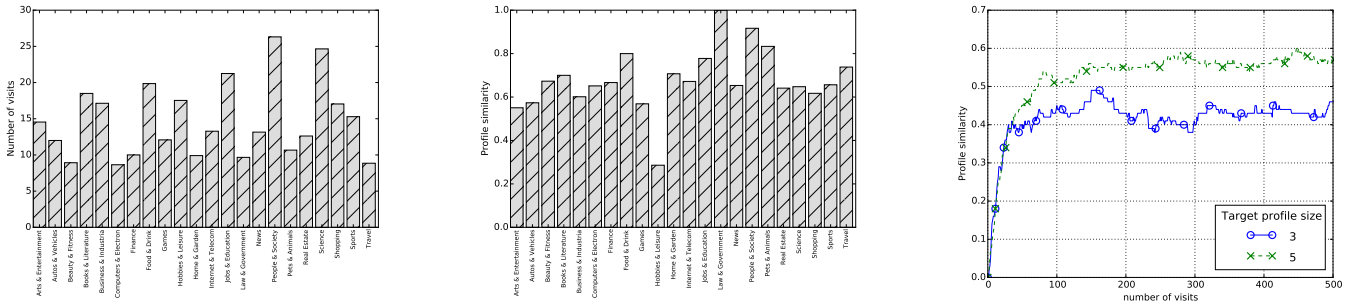
Overall, our experiments indicate that the fake URLs opened by ManTra are a reliable, fast, and accurate way to bias the user profile built by Ads Settings towards the categories of interest associated with those URLs.

V. Conclusions

We have presented ManTra, a user-centered system for web tracking management that presents several unique features.

- It provides the user with the capacity to know what ad networks have likely inferred about his/her interests, as opposed to merely detecting that they are collecting data about him/her.

- It lets the user choose to either improve targeting or better protect privacy, whereas most research and tools focus on either end of the spectrum and do not take into account a user's preferences.



(a) (b) (c)

Fig. 5. Results on Ads Settings. (a) Average number of visits necessary for single-category activation. (b) Average similarity between target profile and Ads Settings profile for single top-level categories. (c) Average similarity between target profile and Ads Settings profile as a function of the number of visits.

- It has the potential to bias the tracking profile built by Google's Ads Settings (and potentially by other ad networks that observe a user's browsing traffic) towards the preferences of that user, without requiring ad hoc infrastructure or assuming the existence of trusted third parties.

Our experiments suggest that ManTra can learn an accurate ad-oriented user profile and is able to influence Google's Ads Settings quickly and effectively.

REFERENCES

[1] M. Backes, A. Kate, M. Maffei, and K. Pecina, "Obliviad: Provably secure and practical online behavioral advertising," in *Security and Privacy (SP), 2012 IEEE Symposium on*. IEEE, 2012, pp. 257–271.

[2] M. Fredrikson and B. Livshits, "Repriv: Re-imagining content personalization and in-browser privacy," in *Security and Privacy (SP), 2011 IEEE Symposium on*. IEEE, 2011, pp. 131–146.

[3] S. Guha, A. Reznichenko, K. Tang, H. Haddadi, and P. Francis, "Serving ads from localhost for performance, privacy, and profit," in *Proceedings of the Eighth ACM Workshop on Hot Topics in Networkis (Hot-Nets)*. ACM, 2009.

[4] V. Toubiana, A. Narayanan, D. Boneh, H. Nissenbaum, and S. Barocas, "Adnostic: Privacy preserving targeted advertising," in *Proc. IEEE NDSS'2010*, 2010.

[5] B. Ur, P. L. Leon, L. F. Cranor, R. Shay, and Y. Wang, "Smart, Useful, Scary, Creepy: Perceptions of Online Behavioral Advertising," in *Eight Symposium On Usable Privacy and Security (SOUPS 2012), Article No.4, Washington, DC, USA*. ACM Press, 2012.

[6] J. R. Mayer and J. C. Mitchell, "Third-party web tracking: Policy and technology," in *Security and Privacy (SP), 2012 IEEE Symposium on*. IEEE, 2012, pp. 413–427.

[7] "Tracking the trackers," http://www.theguardian.com/technology/2012/apr/13/tracking-the-trackers-cookies-web-monitors, 2012.

[8] E. Rader, "Awareness of Behavioral Tracking and Information Privacy Concern in Facebook and Google," in *Tenth Symposium On Usable Privacy and Security (SOUPS 2014), Menlo Park, CA, USA*. ACM Press, 2014, pp. 51–67.

[9] L. Rainie, S. Kiesler, R. Kang, and M. Madden, "Anonymity, Privacy, and Security Online," Pew Research Center's Internet & American Life Project, Washington, D.C., Sept., Tech. Rep., 2013.

[10] A. Acquisti, L. John, and G. Lowenstein, "What is privacy worth?" in *Proceedings of theTwenty First Workshop on Information Systems and Economics (WISE)*, 2009.

[11] N.Smith, "Consumers require value in exchange for data," Marketing-week., Oct., Tech. Rep., 2012.

[12] T.-F. Yen, Y. Xie, F. Yu, R. Yu, and M. Abadi, "Host fingerprinting and tracking on the web: Privacy and security implications," in *Proc. IEEE NDSS'2012*, 2012.

[13] F. Roesner, T. Kohno, and D. Wetheral, "Detecting and defending against third-party tracking on the web," in *Proc. USENIX NSDI 12*. USENIX, 2012.

[14] A. Reznichenko and P. Francis, "Private-by-design advertising meets the real world," in *Proc. ACM CCS 2014*, 2014, pp. 116–128.

[15] A. Siegi, B. Mobster, and R. Burkei, "Web search personalization with ontological user profiles," in *Proceedings of the of the 16th ACM International Conference on Information and Knowledge Management (CIKM 2007), Lisbon, Portugal*. ACM Press, 2007, pp. 525–534.

[16] J.-W. Ahn, P. Brusilovsky, J. Grady, D. He, and S.-Y. Syn, "Open user profiles for adaptive news systems: help or harm?" in *WWW '07: Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada*. ACM Press, 2007, pp. 11–20.

[17] Y. Hijikata, K. Okubo, and S. Nishida, "Displaying User Profiles to Elicit User Awareness in Recommender Systems," in *Proceedings of the 2015 IEEE/WIC International Conference on Web Intelligence, Singapore*. IEEE, 2015, pp. 353–356.

[18] M. Flores and A. Kuzmanovic, "Synthoid: Endpoint user profile control," in *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, vol. 2. IEEE/WIC/ACM, 2014, pp. 242–249.

[19] C. E. Wills and C. Tatar, "Understanding what they do with what they know," in *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society*, 2012, pp. 13–18.

[20] A. Datta, M. C. Tschantz, and A. Datta, "Automated Experiments on Ad Privacy Settings - A Tale of Opacity, Choice, and Discrimination," *Proceedings on Privacy Enhancing Technologies*, vol. 2015, no. 1, pp. 92–112, 2015.

[21] S. Dumais and H. Chen, "Hierarchical classification of Web content," in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA*. ACM Press, 2000, pp. 256–263.

[22] T.-Y. Liu, Y. Yang, H. Wan, H.-J. Zeng, Z. Chen, and W.-Y. Ma, "Support Vector Machines Classification with A Very Large-scale Taxonomy," *IACM SIGKDD Explorations Newsletter*, vol. 7, no. 1, pp. 36–43, 2005.

[23] M. Aly, "Survey on multiclass classification methods," *Neural Netw*, pp. 1–9, 2005.

[24] G.-R. Xue, D. Xing, Q. Yang, and Y. Yu, "Deep classification in large-scale text hierarchies," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 619–626.

[25] S. Gopal, Y. Yang, B. Bai, and A. Niculescu-Mizil, "Bayesian models for large-scale hierarchical classification," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 2411–2419.

[26] M.-Y. Kan and H. O. N. Thi, "Fast webpage classification using url features," in *Proc. ACM CIKM 2005*. ACM, 2005, pp. 325–326.

[27] E. Baykan, M. Henzinger, L. Marian, and I. Weber, "Purely url-based topic classification," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 1109–1110.

[28] C. Carpineto and G. Romano, "A survey of automatic query expansion in information retrieval," *ACM CSUR*, vol. 44, no. 1, pp. 1–50, 2012.

[29] P. Bille, "A survey on tree edit distance and related problems," *Theoretical Computer Science*, vol. 337, no. 1–3, pp. 217–239, 2005.