Semester Project

Tell Me Your Ads and I'll Tell You Who You Are

Alexandra Mihaela Olteanu

EPFL
School of Computer and Communication Sciences, MA2

June 8th, 2012

Supervisor
Nevena Vratonjic
EPFL
CH-1015, Lausanne

Professor
Prof. Jean-Pierre Hubaux
EPFL
CH-1015, Lausanne

Table of Contents

Chapter 1

Introduction

In the past few years there has been a new trend in advertising: Online Behavioral Advertising (OBA), which relies on tracking end users' behavior across web through the websites they browse. This allows Online Advertise Networks, such as Google, to deliver automatic ads that are targeted for an entire history of behavior. Before OBA, ads were automatically targeted to a website content. Such ads are called *contextual ads*, while ads that are targeted on behavior are called *behavioral ads*.

For example, assume that through our past browsing history, we have established that we are interested in finance websites. Sometime in the future we browse some website whose content is sports related. Seeing sports ads on this website would qualify as contextual ads, while finance related ads would be behavioral ads.

While contextual advertising uses only present information about a user - the current website they browse, OBA uses *past* information as well. In fact, OBA is one of the fastest growing and most successful forms of advertising because it is effective: by providing more knowledge in the decision making process of an Online Advertising Network, more relevant ads can be delivered to the end users. The more relevant the ads, the more likely they are to click on them, the more revenue they generate.

Google AdSense, one of the most popular Online Advertise Networks, uses a mix of the two techniques. Their policy is to deliver both contextual ads and behavioral ads, but it is not clear **when** or **how much** of the both they actually do.

In this work, we raise several concerns regarding OBA. First of all, end user have very **little control** over behavioral ads. Once they are in place, it is **hard to predict** what part about an end user's past interests they will reveal. Google does offer an option to optout of receiving behavioral ads, but the **default** is to **optin**, thus allowing behavioral ads. It is a known fact that people are not likely to deviate from the default setting of a service they are offered - take the example of optin versus optout for organ donation [7]. Revealed donation rates were about twice higher when people had to optout of being organ donors, than when they had to optin, i.e. more people are "willing" to donate if that is the default setting. Of course, this might be a case when a default optin setting might have favorable impact - more donations achieved - but with behavioral advertising this is not the case. It is plausible that most of the users are either unaware of the optout option or not using it due to convenience, thus the majority of end users are allowing the tracking of their browsing history and are, in fact, receiving behavioral ads.

Secondly, **security** is a major concern. Although the AdSense code - the code that displays the ads - can be placed on a page using Secure Socket Layers (SSL), Google AdSense does not offer a SSL version for their code, meaning the ads will be delivered without SSL. Lack of SSL means that an attacker (such as an ISP) can intercept the unencrypted traffic and get the end user's ads in plain text. This was not a big thread for contextual ads (an attacker who

intercepts traffic can see the website being browsed and that already give him an idea of the present interest). However, behavioral ads hold more private information because they reveal any of the possible past interests of an end user. Thus, if behavioral ads are retrieved by an attacker, they can reveal a lot more information than contextual ads. Moreover, we show in this work, that the mechanism Google uses to keep track of an end user behavior is easy to hack (as we show in 3.4).

Last, but not least, the lack of security leads to **loss of privacy**. Online privacy is an important right which should be guaranteed to all online users. As [1] shows, 94% of end users consider online privacy important and think about it often, while 29% (respectively 40%) end users think Online Advertise Networks (such as Google) are solely (respectively partly) responsible for online privacy protection. Imagine yourself at the office in front of the computer with colleagues when an ad shows up about leaving work and going back to school, or relocating over seas deals, or worse yet, dealing with alcoholism or other illnesses? So there is no surprise that 54% of end users actually don't like behavioral ads.

Moreover, early this year Google merged the privacy policies for all their products. This means than all the information they can gather about an end user through any of their products - Gmail, Google Docs, search queries etc - can be used to build his preferences profie. Now more than ever, it is interesting to determine exactly what the privacy loss risk is by allowing behavioral ads.

Our goal in this work is then to *understand and quantify the loss of online privacy through OBA*. And, consequently, finding good means of quantifying behavioral ads, which is not a trivial task.

At this time we are not aware of any other work trying to quantify behavioral ads.

Chapter 2

System Model

In this chapter we present background information and describe the setup for our work.

2.1 Google Online Advertising System

Google AdSense is a free platform, to which any website owner can register. By registering, they allow Google to display ads on their website. This yields revenue for both the subscribers to Google AdSense, and of course Google. The ads that are shown on a website are targeted to the site content, as well as end user behavior - the visitors of the website.

Upon registration, a website owner will receive a unique publisher id, which they have to incorporate in the source of their site. They also designate a space on their page where Google Ads are to be displayed and they can select the type of ads they want to show (text, media). They have *no control* over which individual ad is shown.

As a means to provide more control to end users, Google has recently introduced the Ads Preferences Manager [2] - which is an online tool where they allow individual end users to manage and trace their experience with Google Ads. They can here optin or optout of behavioral ads and they can see what Google has saved as their interests at some point.

The means by which Google stores the interests of one end user is through the use of a third party ID cookie. This cookie is specific to each machine and each browser, but, surprisingly enough, not to a Google identity (for example Gmail account). Every time an end user is browsing a website, the Google Ads cookie is associated to the request and the interests associated to his cookie are updated accordingly. For example, when browsing a sports website, an interest of sports may - if not already present - be associated with the cookie. In the event that the visited website supports Google AdSense ads, the Google servers are also queried to retrieve all the specific interests associated with that cookie and then behavioral ads are delivered according to those interests. Figure 2.1 shows a screenshot of the Google Ads Preferences Manager where the user id cookie is emphasized.

The Google Ads id cookie can be seen with the help of any tool that inspects HTTP headers. Figure 3.6 shows an example using Firefox browser and a Firefox AddOn Live HTTP Headers, which shows all the HTTP headers while browsing. We notice that the same cookie id that is displayed in the Google Ads Preferences Manager is attached to the requests **unencrypted**.

2.2 Adversary

We can imagine a scenario where an attacker - any ISP for example - intercepts end user traffic and directly sees the clear text ads - due to lack of SSL.

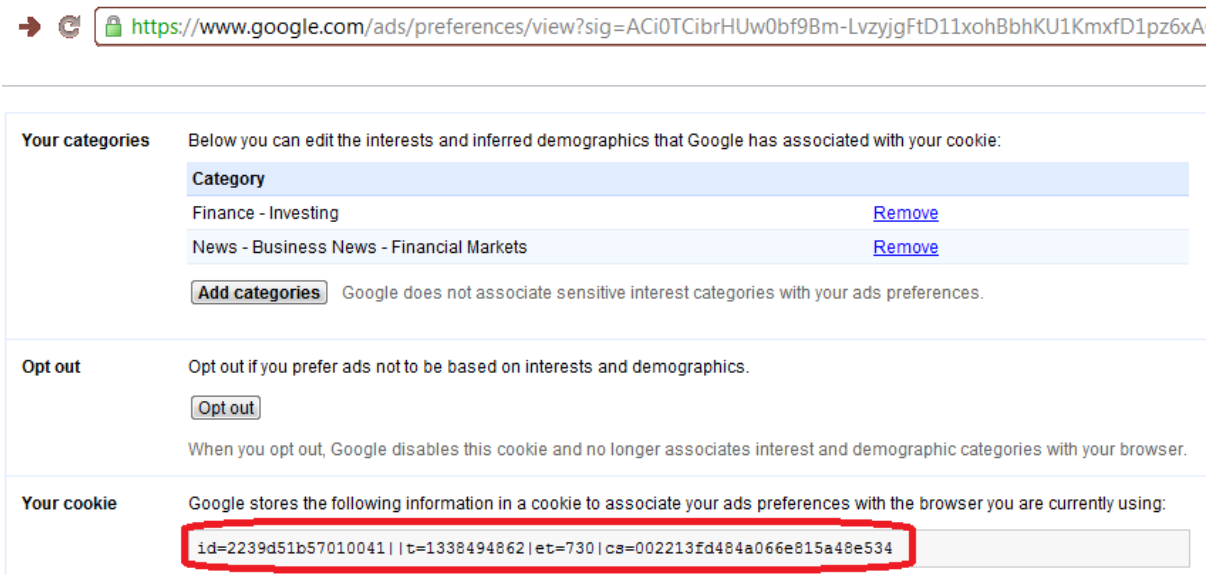


Figure 2.1: A view of the Google Ads Preferences Manager

Alternatively, an attacker could at the very least inspect the HTTP headers from traffic, retrieving the cookie id and then try to use it maliciously. We point out the HTTP headers can be retrieved from a *remote* location with several network traffic monitoring tools. We want to learn what an attacker can do, once it has hold of an end user's cookie id; if he can learn anything from it by retrieving ads or if he can, worse yet, manipulate the cookie.

2.2.1 Passive attack

Firstly, we want to see if an attacker can somehow send a stolen cookie id to the Google servers and retrieve the ads that the end user (the real owner of the cookie) would get. We show in this work that the answer is yes. We call such a scenario when an attacker only wants to **learn the end user preferences**, a **passive attack**.

Through this kind of attack, an attacker can answer different questions about an end user. In our experiments, we build an end user preference profile by browsing websites of categories I_1, I_2, I_3. At this time, we assume an attacker can spy on end user traffic and obtain his id cookie. He then browses websites of another category, C, using the id cookie and retrieves the ads the user would receive, trying to answers questions such as:

- What percent of ads corresponding to I_1, I_2 *or* I_3 does he get (these would be behavioral ads)?
- How long (how many websites) does he have to browse in order to learn all of the end users existing interests (I_1, I_2 *and* I_3)?
- How long until he gets no more ads of category I_1, I_2 *or* I_3 anymore?

2.2.2 Active attack

Secondly, we want to see if an attacker can use a stolen cookie to impersonate the end user and browse websites under this false identity. If so, we want to see if his actions have any effect on the real end user. The answers to these questions, we show, are, again, yes. We discover that

an attacker can attach the end user cookie to requests when browsing websites of his choice, thus causing the interests of the end user to be modified and, in effect, influencing the ads the end user is getting. We call this type of attack, an **active attack** because the attacker has the power to **change the user preferences**.

In this scenario, we can imagine that the attacker retrieves the id cookie and then wants to enforce some type of interest, I to the user, in order to make him see a certain type of ads. For example, assume the attacker is an ISP that wants to push car related ads because they have connections to a car selling business. They could just browse car related websites, under the end user identity. We want to see:

- Does the Ads Preferences Manager of the end user actually get modified by remote browsing with their stolen cookie?
- If so, does the user then get I type ads? And how many, how soon after the attack?
- How long until he doesn't get I type ads anymore? In other words, how often would the attacker have to repeat the attack to ensure an end user constantly gets their ads?

Chapter 3

Methodology

3.1 Building a dataset: collection, validation and categorization

First and foremost we need to build a dataset by *collecting* websites that support Google Ads. We look at popular websites, as deemed by alexa.com - a popular provider of global web metrics. It provides a globally ranked list of the top 1000000 websites in the world [3], based on traffic (which we will call Top1M). We believe searching among these top is good because, being popular, they are likely to show ads.

We additionally want our websites to be *categorized* in order to make associations with the categorization done by Google as displayed via the Ads Preferences Manager. The entries in the Top1M are only ranked, but not categorized, so we also look up an additional listing Alexa offers: the categorized listing [4]. This is a classification of websites into categories, subcategories, subsubcategories etc, with a locally ranked list of a fixed number of websites for each individual category. There are 17 root categories, which we refer to as level zero categories: {"Computers","Sports","Arts", "Science","Business", "News", "Recreation", "Games", "Health","Society", "Shopping", "Kids_and_Teens", "Home", "Reference", "Regional", "World" , "Adult" }. Each of these has a varying number of subcategories - which we will call level one categories. For example the top category Arts has direct subcategories {Animation, Architecture, Art History, Awards, Bodyart, ..., Visual Arts, Writers Resources}). Each of them can have a varying number of 'level two' subsubcategories. Arts/Animation has subsubcategories {Anime, Artists, Audio, Awards, Cartoons, ..., Voice Actors, Writers}) and so on reccursively. The depth of this categorization tree can vary for every leaf (Arts does not have to have as many levels of subcategories as Shopping) and for categories the number of subcategories can also vary across one level.

At any time in this tree of websites for a certain subcategory, Alexa offers a *ranked* local listing of *at most* 500 top websites. The ranking is *local* to the specific subcategory, which is not quite what we want.

The first difficulty we encounter is that there is no way to infer from the locally ranked hierarchy of websites the global ranking of a website, while Top1M contains the global ranking of a website and the website, but there no category information associated to it. In order to overcome this, we will try to crossreference these two listings in order to obtain a *globally ranked categorized list* of websites.

We first parse the categorization tree by crawling the Alexa website. Because of the high variation in number of levels, branching into subcategories for every category in a level, as well as number of websites in each local listing, it is difficult to estimate how many levels we need to inspect in this tree in order to ensure that, in the end, we will have visited at least 1000000 websites. We have selected a maximum level of recursion of 3 (this means we look at

Arts/Animation/Anime/Clubs_and_Organizations for example, but not further down).

In practice, we found that we inspected in total 1610947 websites. **Notice that this satisfies our need to inspect at least 1.000.000 websites: the union of a *larger* set of locally top ranked websites ensure that the globally top ranked will indeed be inspected**. By the same logic, we point out that if any of the websites we collected through crawling that support Google Ads is not assigned a global rank after cross-referencing with the Top1M global rank list, it must mean that its global rank would be higher than one million. They might still be useful for large scale experiments, but for now we only keep those we could rank.

We *are only interested in websites that support Google Ads*, but we have no intuition about how many do - we don't know of related official statistics or work, so we believe this will be a contribution of our work. Among all the websites we inspect, we only keep those that support Google Ads and also appear in the Top1M because we can get a global ranking for them. We describe the mechanism by which we determine if a website supports Google AdSense ads in 3.3.

Categorization of websites, as well as ads, is a crucial tool that allows us to further study ads as behavioral or contextual. Thus we need a mechanism to categorize both ads and websites. For websites, Alexa's hierarchical categories already offers some direction, but we do not know how well this maps to what Google does internally. For categorizing textual ads we will use the external Alchemy API [5]. We believe categorizing websites with Alchemy API too, will provide some consistency, as well as give more certainty to the website categorization as given by Alexa.

Specifically, for websites, we use two categorization sources (Alexa and Alchemy), from which we want to derive a categorization that is most likely to be similar to the one that Google does internally (as seen in the Ads Preferences Manager). In order to achieve this, we filter only websites for which Alexa and Alchemy both agree (for example Alexa's "Arts/Animation/Anime/Clubs_and_Organizations" or "Arts/Animation/Movies/Titles" would both be matched by Alchemy's "arts_entertainment" category; Alexa's "Sports/Airsoft/Teams" would be matched by "sports" and so on).

We identify eight categories **{computer_internet, sports, arts_entertainment, business, science_technology, recreation, gaming and health}** for which there is a *good match* between Alexa and Alchemy categorization (as seen in Table 3.1). A category is well matched if the two mostly agree. We will call these eight categories **Target Categories (TCs)**.

Requiring that both categorizations match and overlap in high percentage is one way of *validating* our categorization. As an additional **validation** step for website categorization, we also take into account of the fact that Alchemy API returns, for each categorization, a confidence score between [0, 1] which shows how confident they are on the given categorization (higher is better). We only keep, for each target category, those websites for which Alchemy yields a confidence score higher than the average confidence score within that target category.

Section 3.6 will show an additional validation step for our target categories.

Summarizing our technique, for each website, our data gathering steps are:

1. collect website and category from Alexa
2. keep website if it supports Google Ads
3. get a second categorization from the Alchemy API (category and confidence score)
4. assign to one of the target categories as determined by both Alexa and Alchemy categorization

5. check and keep if a global rank from Top1M can be assigned to it

Figures 3.3, 3.4, 3.1 and 3.2 show different visualizations for our data selection process. 109.670 of all the websites inspected with the Alexa (1.610.947 in total) categorized tree parser support Google AdSense ads, yielding a 6.81% rate. Eventually, over our 8 target categories, we kept 15.538 websites, 2.632 of which can be globally ranked in the Top1M.

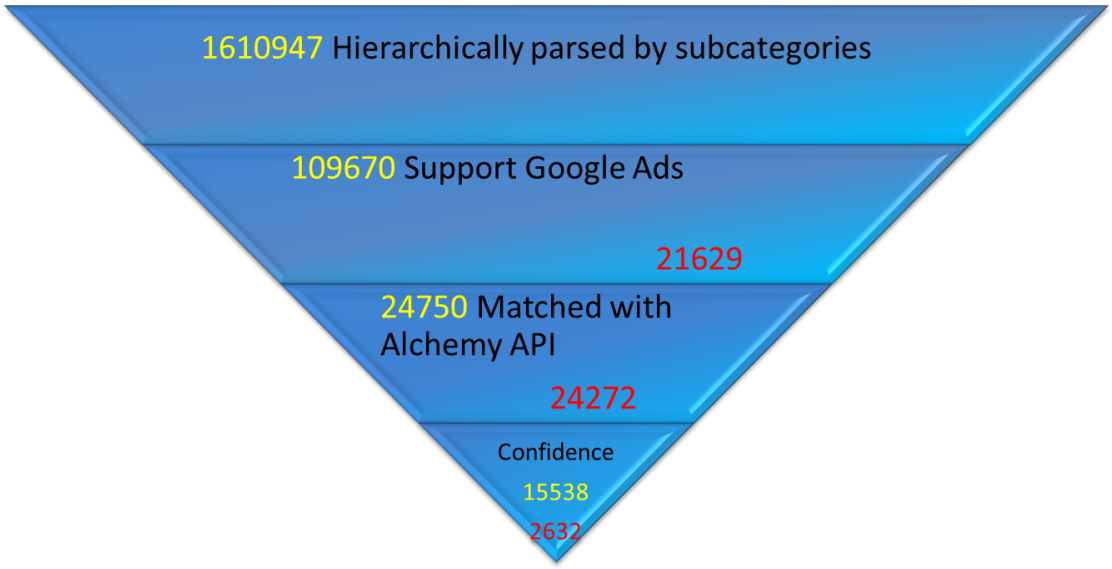


Figure 3.1: Size variation of the dataset while applying successive filters. The red numbers represent the respective subsets that were successfully ranked globally by cross-referencing the Top1M.

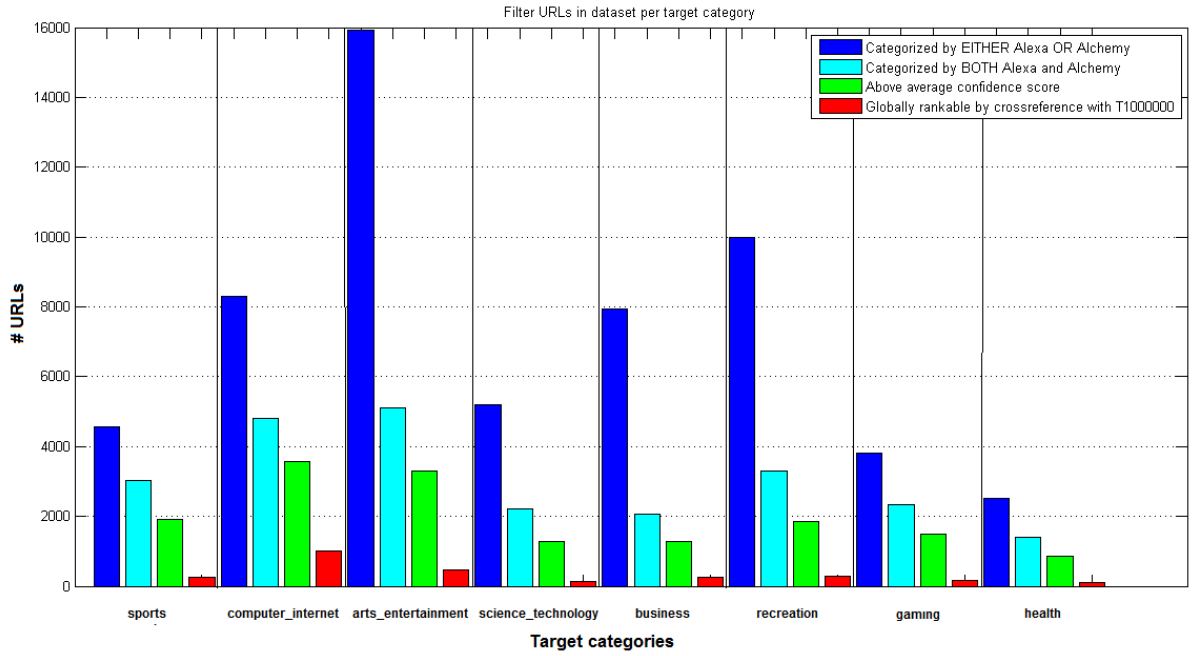


Figure 3.2: Size variation of revelant dataset during the formation of our eight Target Categories

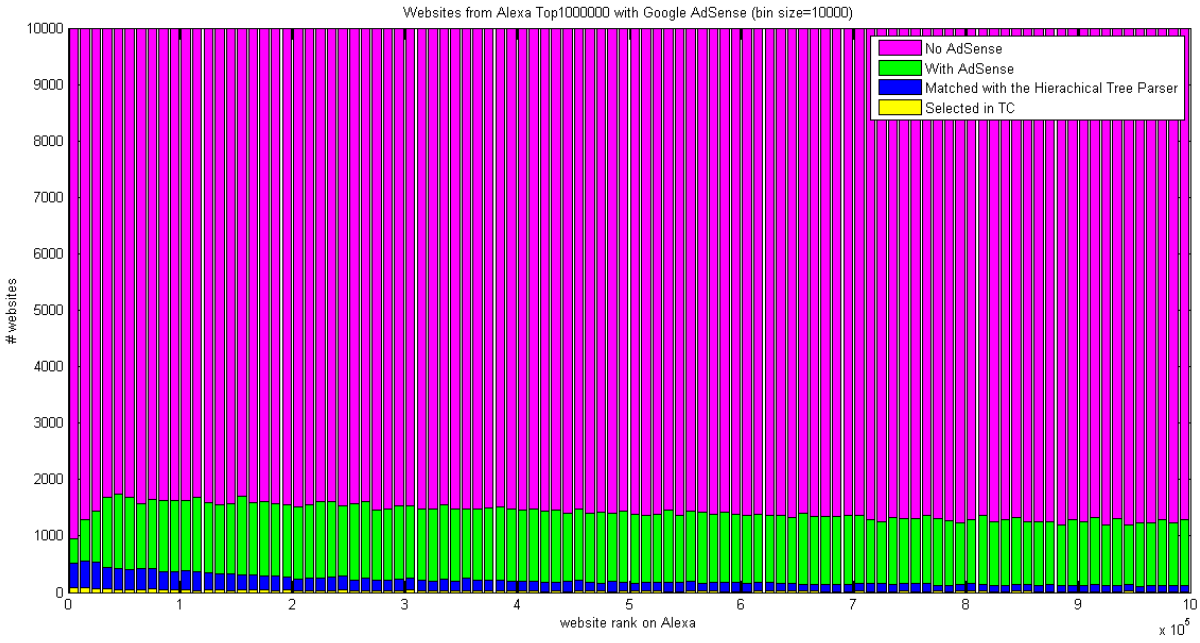
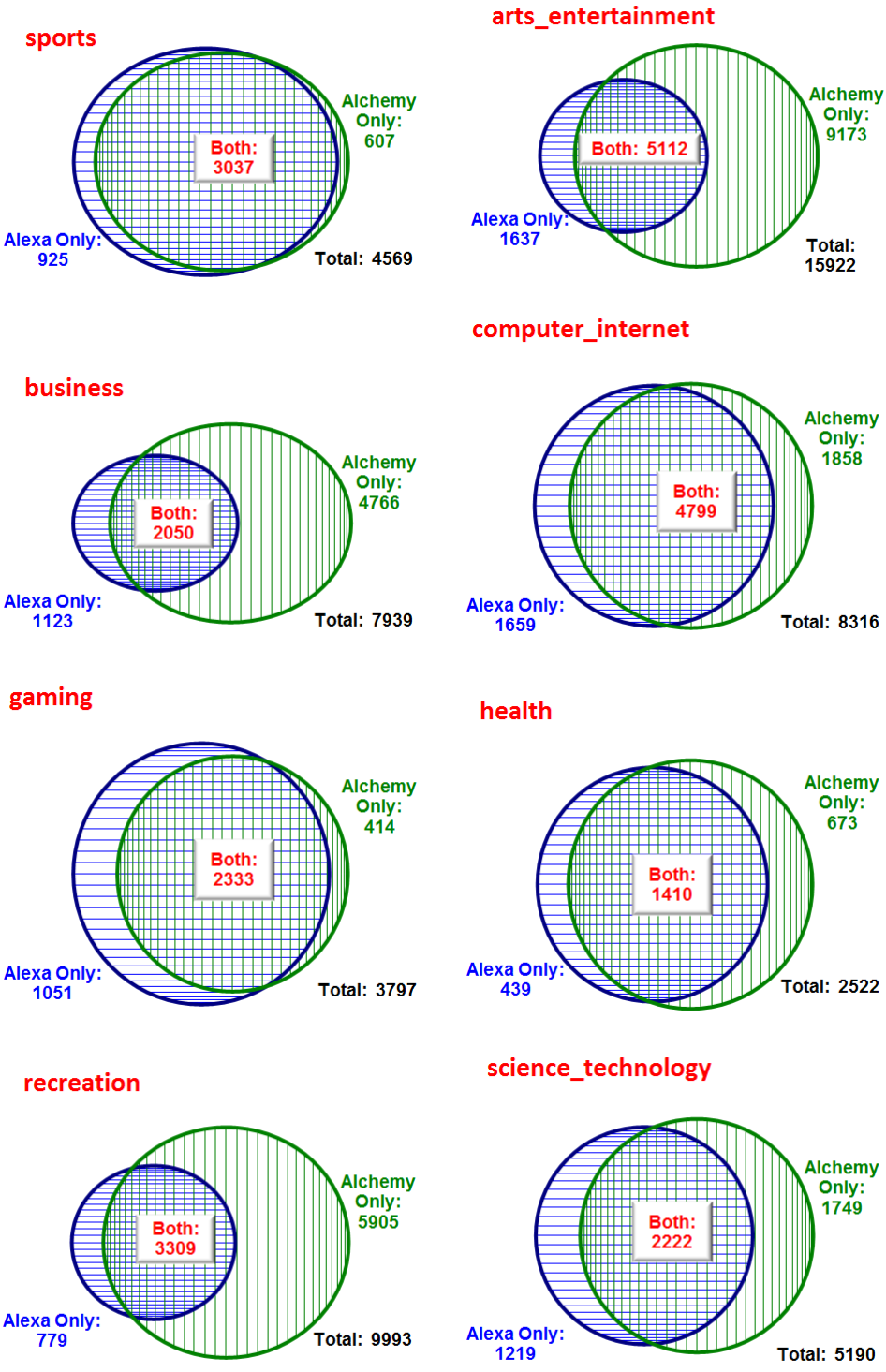


Figure 3.3: Ranking distribution of selected websites for Target Categories.

Table 3.1: VennDiagrams for Target Categories.



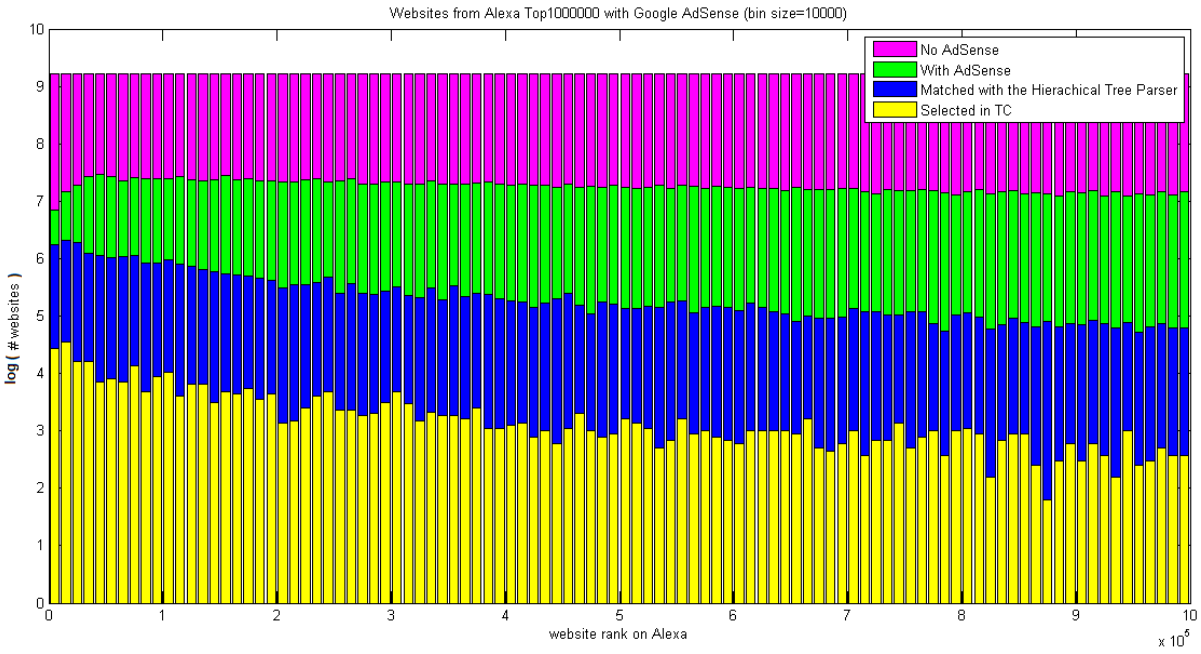


Figure 3.4: Ranking distribution of selected websites for Target Categories (log scale)

3.2 Properties of the Dataset

As an independent validation experiment, we also run a job to identify and mark all the websites in Top1M that support Google AdSense ads (from the uncategorized globally ranked list). It is interesting to note that in the Top1M, the percentage of AdSense supporting websites is higher, 14.22% on average, than it was in the top 1.610.947 which we parsed with the hierarchical tree parser - 6.81% on average. Therefore the distribution of the percentage of AdSense supporting websites in the top X, when X slowly grows to 1.000.000 is worth analyzing. Figure 3.5 shows this distribution, grouping websites in bins of size 3000. For the first bins of most popular websites it is much lower than the average:

- In the top 500 - 4.2%
- In the top 1000 - 4.2%
- In the top 2000 - 5.25%
- In the top 5000 - 7.7%

The figures in Table 3.2 give an idea of the individual spatial distribution by rank of the top websites supporting Google AdSense. *The top 108 most popular websites do not support Google AdSense.* For the first few thousands of popular websites the percentage of websites with Google AdSense is much smaller than the average. We then observe a rapid increase, to a peak of almost 16% (between ranks 150.000 and 300.000 - bins 50-100), followed by a very slow decrease. A possible explanation for this behavior is that popular websites have their own advertising units and deal with advertisers themselves. Ad sense is better suited for the "heavy tail" websites, as it allows cheap, automatic ad inclusion.

Given the lower average we obtained in the extended set of hierarchically parsed data (6.81%), we think it is probable that a slow decrease would continue beyond the top 1.000.000 threshold. The reason is that the set of websites from Top1M with AdSense should be included in the larger set of websites with AdSense from the larger, top 1.610.947, set. It would be interesting to test this intuition, if we had a larger official dataset of globally ranked websites.

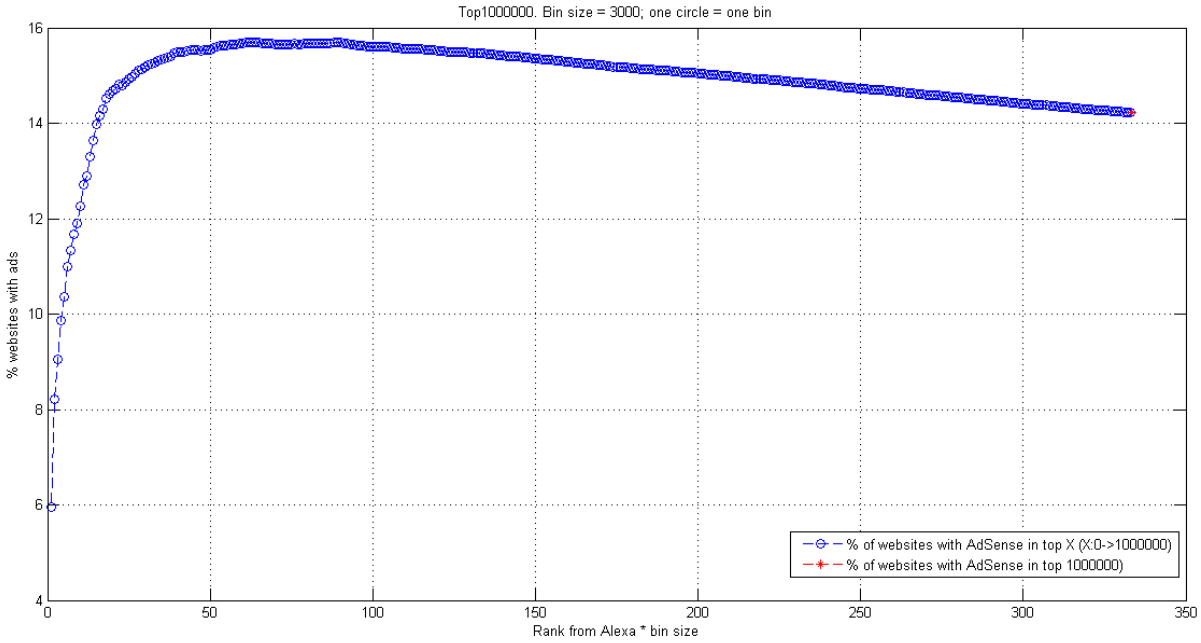
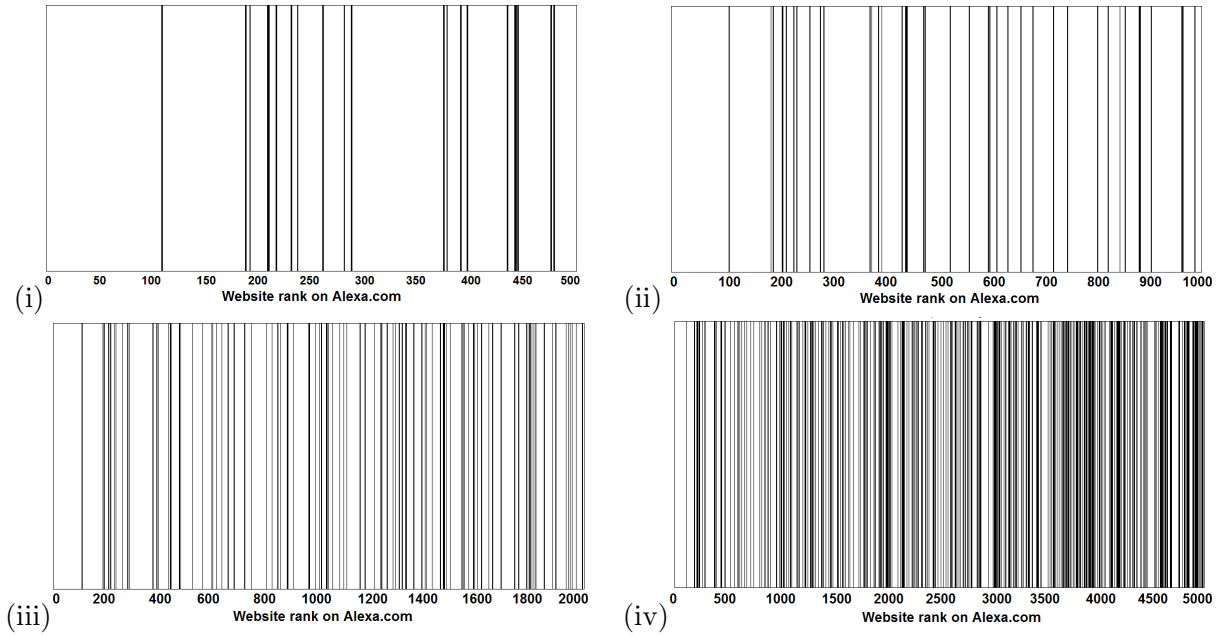


Figure 3.5: Distribution of the percentage of AdSense supporting websites across the Top 1000000. The average is in red.

Table 3.2: Distribution of the AdSense supporting websites among the top (i) 500, (ii) 1000, (iii) 2000, (iv) 5000 websites. A black line, respectively white line, represents an AdSense supporting, respectively not supporting, website.



3.3 Determining if a website supports Google AdSense ads

The official code standard declared by Google to include AdSense ads on websites, is to call a javacript http://pagead2.googlesyndication.com/pagead/show_ads.js and give it as parameter the published ID, which is assigned to each website owner when joining the AdSense program. However, by inspecting the page sources of multiple websites in the Google Display Network, which is a list Google publishes of websites that use Google AdSense, **we noticed some variation from the standard**, and we've incorporated **four patterns** to search for when extracting the publisher ID (details on the patterns can be found in the Appendix A.1).

For a certain website, we extract the page source and we look for a publisher ID using all the possible four patterns. If a publisher ID is found by any of these patterns we declare that the website supports Google AdSense ads.

3.4 Retrieving ads

In order to answer our research questions, we look only at text ads at this time, which are easiest to analyze.

Finding a way to retrieve the ads that Google would display on a certain website to a certain end user (identified by an id cookie) is one of the most challenging part of our work. We need to fully understand how the Google AdSense code works, from the published ID that can be found in the source of any ad supporting website, to the actual ads. Notice that although the *ads* can be seen in browser, they *are not directly visible in the source page*. The typical process is that the previously mentioned javacript show_ads.sj is called with published ID as parameter. Studying the source code of the different scripts that are called, and tracking all the HTTP requests that are sent before a website is actually displayed in the browser, we determine that in order to retrieve ads, a call to "http://googleads.g.doubleclick.net/pagead/ads?..." is enough and it emulates the Google natural process, providing the following parameters:

- **client** - the published ID of website that is making a request to display ads
- **format** - the format in which to display the ads (one per page, side column with 6 ads etc)
- **ad_type** - the type of ads, for which we use 'text'
- **url** - the requesting url (of the original website requesting ads)

In the natural process, the request that is made through the show_ads.js script sends many more specific parameters, but through extensive testing, we determined this set of four is the minimal set needed in order to guarantee that the same ads are obtained by calling the googleads.g.doubleclick.net URL as they would be by manually browsing the requesting URL.

We mention here that the typical number of text ads that can be displayed on a website at one time ranges from 1-6, and they randomly vary between different visits of the same website. Thus, in order to get a **representative sample** of ads that could be displayed on a website, we repeat the call to thegoogleads.g.doubleclick.ne URL mutiple times (10 in our experiments) and only keep unique ads.

In order to be able to implement the attack scenarios we described, a random number of websites and possibly random set of websites would need to be browsed in an *automatic fashion*. In order to do this, it is enough to make the call to the googleads.g.doubleclick.net URL with the right parameters and attach the id cookie to the request (our code base was written in Java).

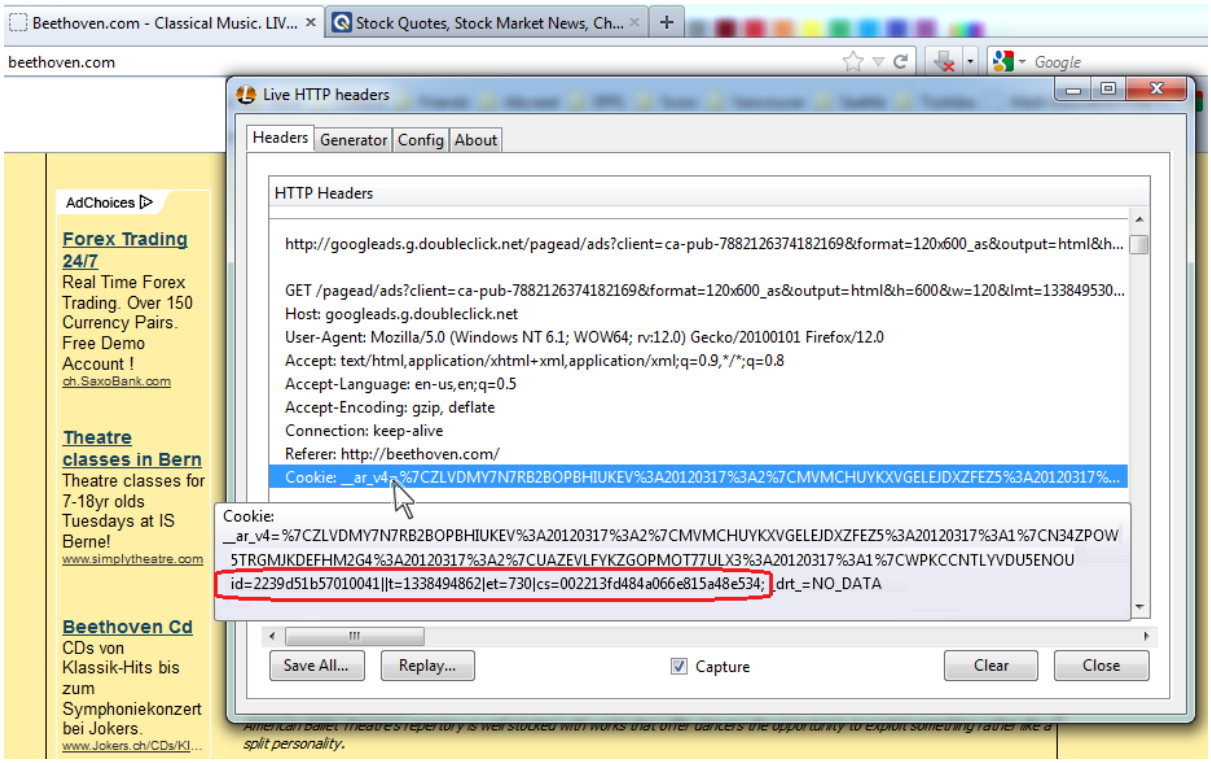


Figure 3.6: Live tracking of http headers. Notice the Google id cookie in plain text, and the Google Ad Sense ads on the left side. They are displayed on beethoven.com, a classical music website. The first ad seems to be about finance and indeed, we did browse a finace related website just before. This is thus an example of a behavioral ad.

Finally, we confirm that through this **automatic browsing of websites** done from **another machine than the end user's machine**, the **Ads Preferences Manager of the end user gets updated with new interests**, in the same way it would through local browsing.

3.5 Categorizing ads

A text ad consists of three elements:

- *title*
- *content*
- *link*

For the simple example of the first ad shown on the website on figure 3.6, these would be: "Forex Trading 24/7", "Real Time Forex Trading. Over 150 Currency Pairs. Free Demo Account !" and http://www.ch.SaxoBank.com, respectively.

In order to decide if an ad is contextual of behavioral, we first need to know what category it has. We use Alchemy API, which offers the feature of returning a category for any text or any URL, to which it also associates a confidence score. We vary the type of information we provide Alchemy API for an ad, in order to obtain a **list of possible categorizations** of the ad. Consequently, we obtain a possible categorization for an ad based on the ad title

alone, the ad content alone, the ad link alone, and based on combinations of any two or three of these ad elements. For each of them, we are offered a confidence score, and we sort all the different categorizations in descending order of this score. If the same categorization is obtained by different criteria, we keep the one with highest confidence score. As an example, for the previous mentioned ad, these are the categories that we obtain in the final list:

- business - with confidence score 0.925712
- computer_internet - with confidence score 0.442094

From this **list of possible categorizations**, we call the **most likely category** of an ad, the one with the highest confidence score or the one which coincides with the url category the ad is displayed on - if this exists. For example, if the ad above is displayed on a sports url, its most likely category will be the highest scored one - business, but if it is displayed on a computer_internet url, then that will be its most likely category. This is a notion we will use when determining if an ad in behavioral, so we want to be very conservative when saying if an ad is indeed behavioral, in order to provide a lower bound for the number of behavioral ads.

3.6 Ad baseline for URL category

As we explain in section 3.1, categorizing URLs and ads by a similar logic Google does internally is not easy to do. We obviously have no access to their internal logic, so we need to find other ways of categorization that are as close as possible *in effect* to the Google way. This is reason we use both Alexa and Alchemy to categorize and focus on high confidence score URLs.

In order to *validate* how good our URL categorization is, we perform what we call a **baseline test** for each of our 8 target categories of URLs. Starting with *no existing interests*, given a URL category C, we will browse a set of URLs of that category and count how many ads of each type we receive. We count each ad only once for one URL, according to its *most likely category*. The key idea is that our categorization is good if there is a distinctive peak for ads of category C - matching the URLs category. Because there are no prior interests, most of the ads should be contextual.

We take into account two ways of starting with an empty profile, one by *optout* of the Ads Preferences Manager, and the other by *optin*, but with an *empty* list of existing interests. Figures 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13 and 3.14 show the baselines for each target category of URLs, while Figures 3.16 and Figures 3.15 show these baselines for each setting (optout or optin). In all cases we see distinctive peaks for the ads of category matching the URLs category.

One key observation is that roughly **one third of the total number of ads are received when a user is optout of the Ads Preferences Manager, than when he is optin, but has no interests yet**. So more ads are delivered when opting in. However, in both cases the *percetange* of ads of each category remains roughly the *same* between these two settings.

3.7 Quantifying ads

3.7.1 Contextual ads

We will consider an ad to be **contextual if its *most likely category* is the same as the category of the website it is displayed on**.

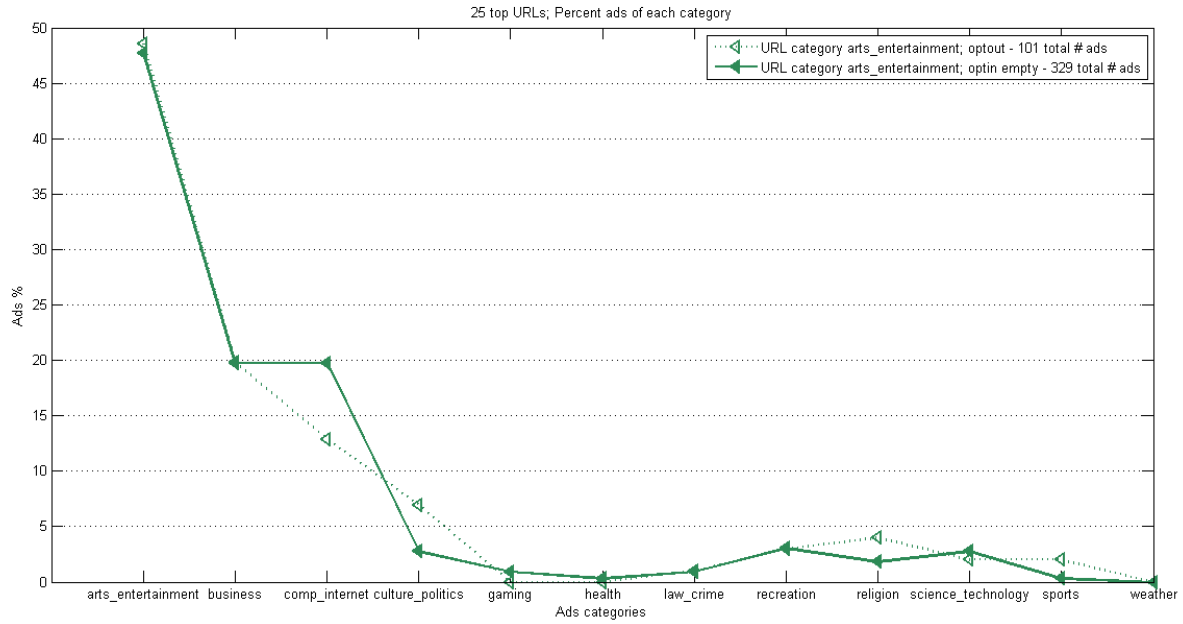


Figure 3.7: Ad baseline for URLs of category arts_entertainment

3.7.2 B2 measure for behavioral ads

Determining if an ad is behavioral is not a trivial task, it is not enough that it is not contextual. Through Google Ad sense, other types of ads are delivered, for instance based on demographics, so we need to take into account the past interests of an end user when determining if an ad is indeed behavioral.

Specifically, given $I_1, I_2, ..., I_k$ - existing interests of an end user, and C - the category of the current website being browsed, an ad **is B2 behavioral if *any* of its categorizations is the same as *any* of the prior interests** $I_1, I_2, ..., I_k$. For example, assuming existing interests business and arts_entertainment, if the ad in our example above 3.5 is displayed on a sports website, than we can say that it is behavioral (because one of its categorizations is business and this matches one of the existing interests). However, if the same ad is displayed on a computer_internet website, than the ad could be both behavioral and contextual (one of the categorizations matches one of the prior interests but also one of the categorizations matches the URL category on which the ad is displayed). In such a case when an ad can be both behavioral and contextual, we call the ad **ambiguous**.

Our **B2 measure for behavioral ads** thus takes into account all these ads that **could be behavioral**; in other words, all the ads for which one of the possible categorizations matches any one of the existing interests. This *includes ambiguous ads* as well, so it is a **relaxed measure** and intuitively it will provide an *upper bound* on the quantity behavioral ads. The measure will effictively return the percentage of such ads from the total number of ads received.

3.7.3 B1 measure for behavioral ads

On the more **restrictive** side, we define another measure for behavioral ads, **B1**, in the following way:

Case a) Assuming only one existing interest I and current URL category C (I <> C):

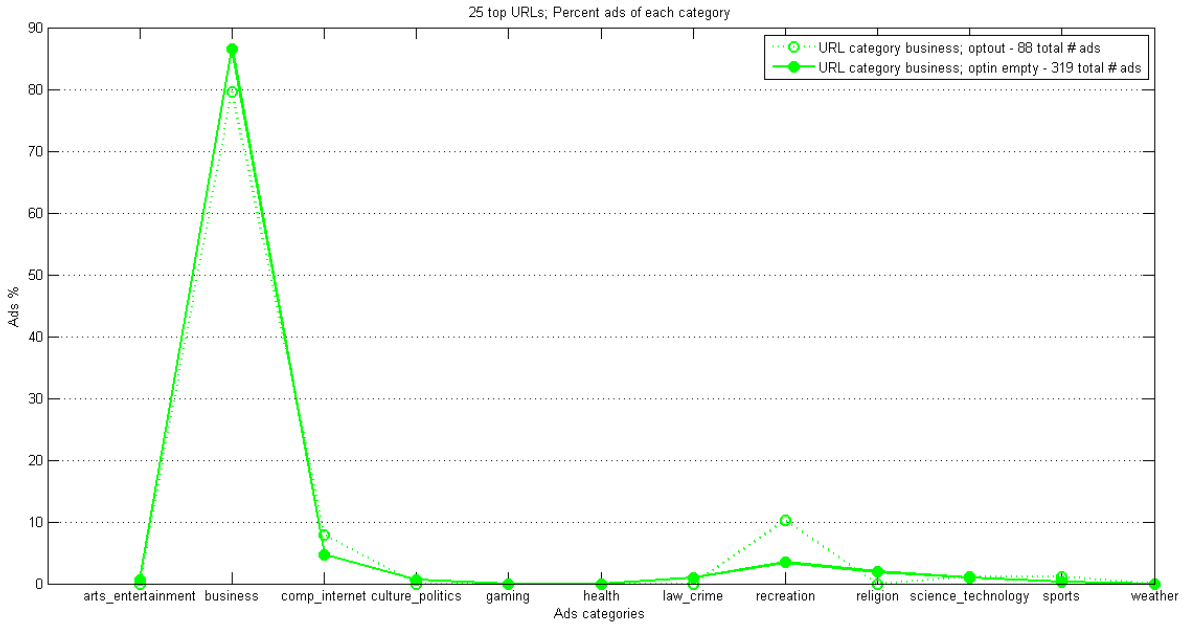


Figure 3.8: Ad baseline for URLs of category business

1. count all the ads for which the *most likely category* is the same as I
2. computer the percentage of these ads, over the total number of ads received
3. from this percentage, *subtract the corresponding percentage of ads of type I that were received in the baseline of URLs of category C.*

Case b) Assuming more existing interests $I_1, I_2, ..., I_k$ and current URL category C (I_1 <> I_2 <> ... <> I_k <> C):

1. for all i=1:k, compute $B1_i$ as described in Case a), assuming only one prior interest I_i
2. compute B1 as $\sum_{i=1}^{k} B1_i$.

By only taking into account one possible categorization for an ad - the most likely one, which was favoring a contextual verdict for an ad if one was possible - and by removing the percentage of ads of type I that were received in the baseline, this measure B1 should intuitively give a *lower bound* on the quantity of behavioral ads. For an example of B1 computation for one existing interest, see A.2.

3.8 Experiment Setup

In this section we describe the type of experiments we run to answer the specific questions we described in section 2.

3.8.1 Building a Preference Profile

In all our experiments, we first need to build a preference profile, i.e. establish one or more existing interests. In order to establish an existing interest I, we browse 25 websites of category

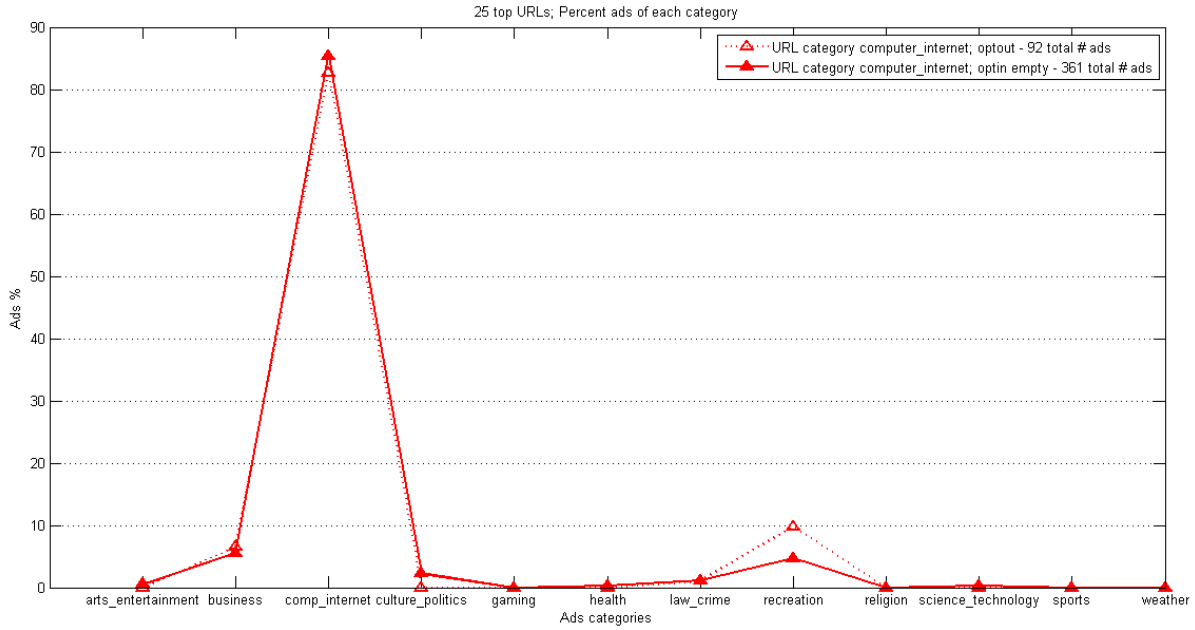


Figure 3.9: Ad baseline for URLs of category computer_internet

I. By manually veryfing the Ads Preference Manager, we determine that this is sufficient for the interest to be established.

In order to establish more existing interests $I_1, ..., I_k$ for the same experiment, we browse URLs from categories $I_1, ..., I_k$ in a *random* order, to ensure that none of the interests gets more weight. If we were to browse URLs of category $I_1, ..., I_k$ sequentially, I_j would more likely be saved in the Ads Preferences Manager than I_i, for any $i, j = 1, k;\ j > i$.

We always choose websites in a category that have the highest Alchemy confidence score in their categorization, to ensure that we are most likely to match the categorization Google would internally attribute to them.

3.8.2 Basic Experiments with One Existing Interest

The logical initial setup for an experiment is to have only one existing interest and by browsing URLs of a given category, collect all the ads that are retrieved. In the trial version of the API, Alchemy only allows a limited number of calls per day, so we limit our experiments accordingly. *Immediately* after establishing an existing interest (arts_entertainment, computer_internet, gaming *or* recreation), we browse 25 URLs of one of the 8 target categories and retrieve the ads. Notice we use all the 8 target categories to retrieve ads, but time constraints did not permit completing these experiments with prior interests science_technology, sports, business or health as well. These could be part of future work. We choose websites in a category that have the highest Alchemy confidence score in their categorization, to ensure that we are most likely to match the categorization Google would internally attribute to them.

These experiments are meant to give an understanding of how behavioral ads are delivered for different combinations of existing interest-current URLs category.

For each experiment we run (of this as well as future types), we save all the ads and their possible categorizations in a database on which we run statistics to compute the measures defined in section 3.7. The ad baseline we use to compute the measure B1 is the optin with

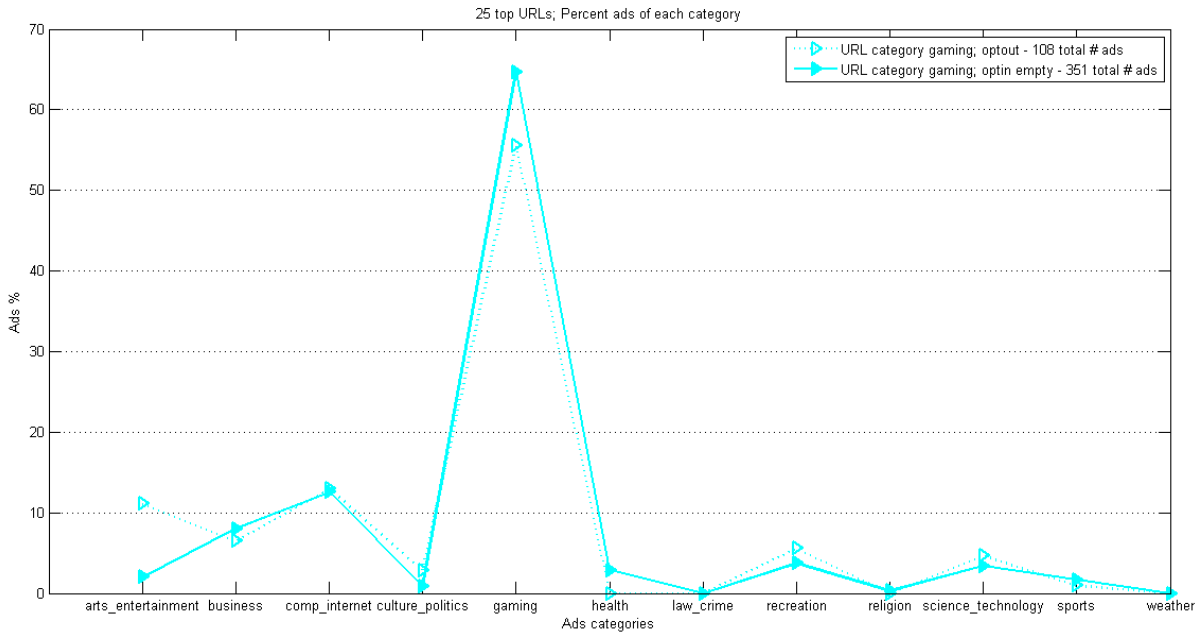


Figure 3.10: Ad baseline for URLs of category gaming

empty preference profile baseline because this offers a higher number of ads than the optout baseline, and, as we will see, this is also similar to the number of ads we receive when having existing interests already established.

3.8.3 Experiments with More Existing Interests

We also run experiments with more existing interests. Specifically, we build a set of prior interests consisting of all the four prior interests we studied in the basic experiments *at the same time* - arts_entertainment, computer_internet, gaming *and* recreation. Immediately after building the preference profile, we browse *200* websites of a current URL category for which we collect ads. As current categories, we use the remaining four possible target categories: health, sports, business and science_technology. Ads statistics are collected in a similar way as in the basic one interest experiments.

These experiments are meant to give an understanding of how ads distribution changes when more existing interests are present and compare this to the one existing interest scenario. They are also helpful in understanding how the ads distribution is changing with the number of websites browsed (notice these are long experiments - 200 websites for ad collection).

3.8.4 Long Experiments - one or more existing interests

The setup described so far allows us to answer ads quantifying questions. In order to answer more detailed questions about the evolution over time of the ads, we perform longer experiments, where we browse nearly 200 URLs to collect ads, *either with one or more existing interests* (as already described in 3.8.3). These type of experiments portray what we call the **immediate evolution** of the behavioral ads (as was the case of all the above experiments).

In this scenario, we first build the preference profile and immediately after, we start browsing websites to collect ads. We try to browse as many websites as we can, without exceeding the daily limit of Alchemy API (this typically leads to at most 200 websites). We want to determine

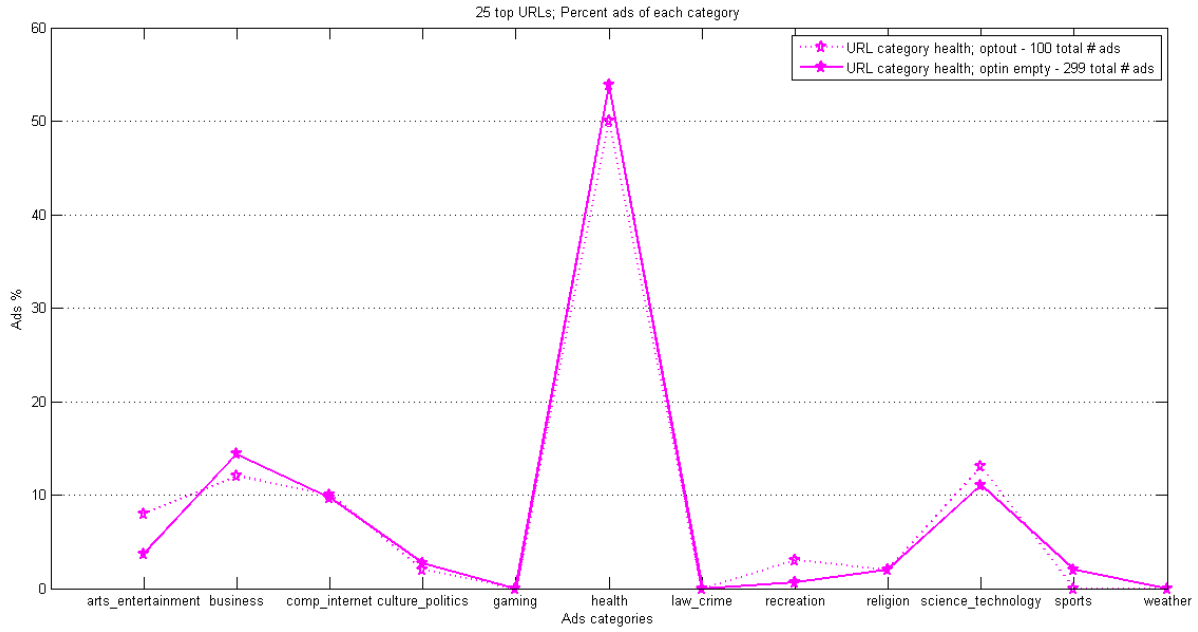


Figure 3.11: Ad baseline for URLs of category health

for how many websites, existing interests stay valid (are exhibited through behavioral ads), as well as how soon they are revealed (how many websites need to be browsed before they are revealed).

We say an **existing interest** is **surely revealed** when its B1 measure is positive - i.e. when more ads of that type are retrieved than in the baseline. We say an interest is **surely lost** when the B1 measure for related ads becomes negative. All of more existing interests are completely revealed, when the corresponding B1 measures for each of them are all positive.

3.8.5 Basic Experiments with One Existing Interest - after 12 hours

Another type of experiment to describe evolution of ads over time is when we let more time pass after the profile has been built before we start collecting ads. We noticed in our tests that the Google Ads Preferences manager very quickly deletes interests that are no longer validated by present browsing. We show results from one experiment with existing profile consisting of interest computer_internet and URLs of type business for which ads are collected. We compare the types of ads we get by running this experiment immediately after the profile has been established, and then again 12 hours later *without changing the preferences profile* in the meantime.

This kind of experiments are meant to prove our empirical observation that interests are deleted - if no other activity - in a few hours. The existing interest is said to be lost if its B1 measure is negative.

3.9 Deciding if a website supports OBA

Of the websites that do support Google Ad Sense, it is also interesting to determine which deliver behavioral ads. Not all websites deliver the same amount of behavioral ads.

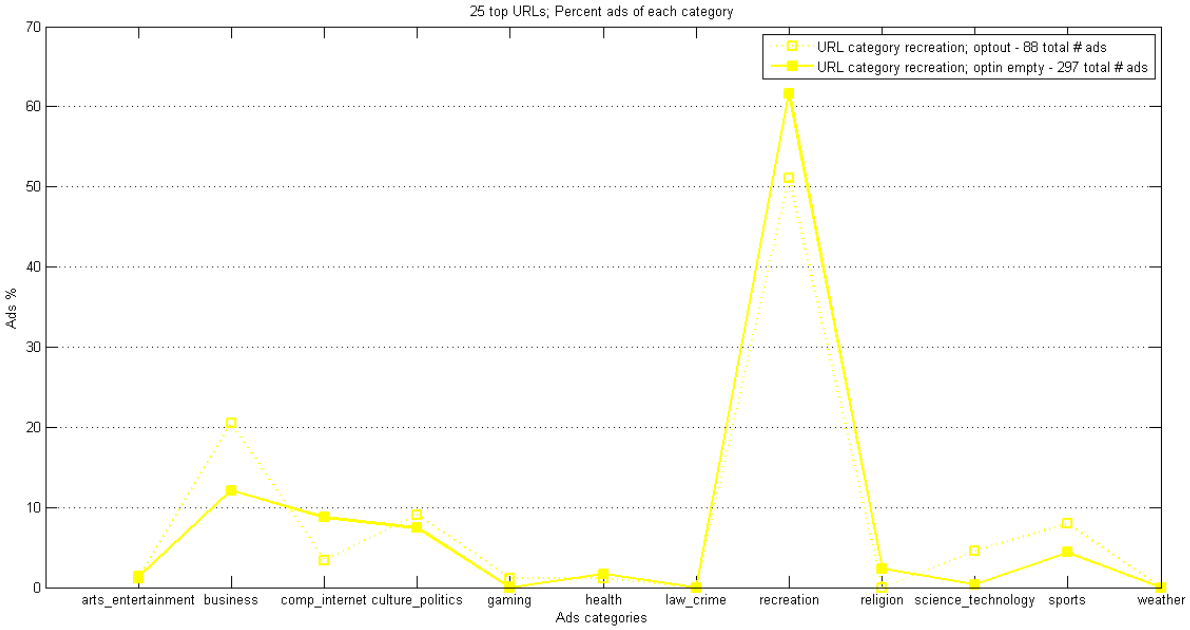


Figure 3.12: Ad baseline for URLs of category recreation

For a certain category C of URLs, we look at all its experiments with one existing interest. We say that a URL of category C delivers behavioral ads of type I, if in the experiment with existing interest I and category C, there is **at least one ad among the ones it delivered that is behavioral with respect to type** I. Since we use two measure for behavioral ads, $B1$ and $B2$, we will also have two measures to specify if a URL supports behavioral ads, for each existing interest I we look at.

To provide more confidence to our two measures, we finally perform a logical AND among the measures for each existing interest and thus we obtain two means of determininvg is a website supports behavioral ads. Specifically, given our four existing interests I_1, I_2, I_3 *and* I_4, we say a website delivers behavioral ads according to measure $B1$ if it delivers behavioral ads of type I_i, for i=1:4. And similarly for $B2$.

We finally count all the websites of our target categories that provide behavioral ads by the two measures. The count we obtain using the more strict measure of behavioral ads, $B1$, will provide a **lower bound** on the number of websites supporting behavioral ads, while the count using $B2$ will provide an **upper bound**.

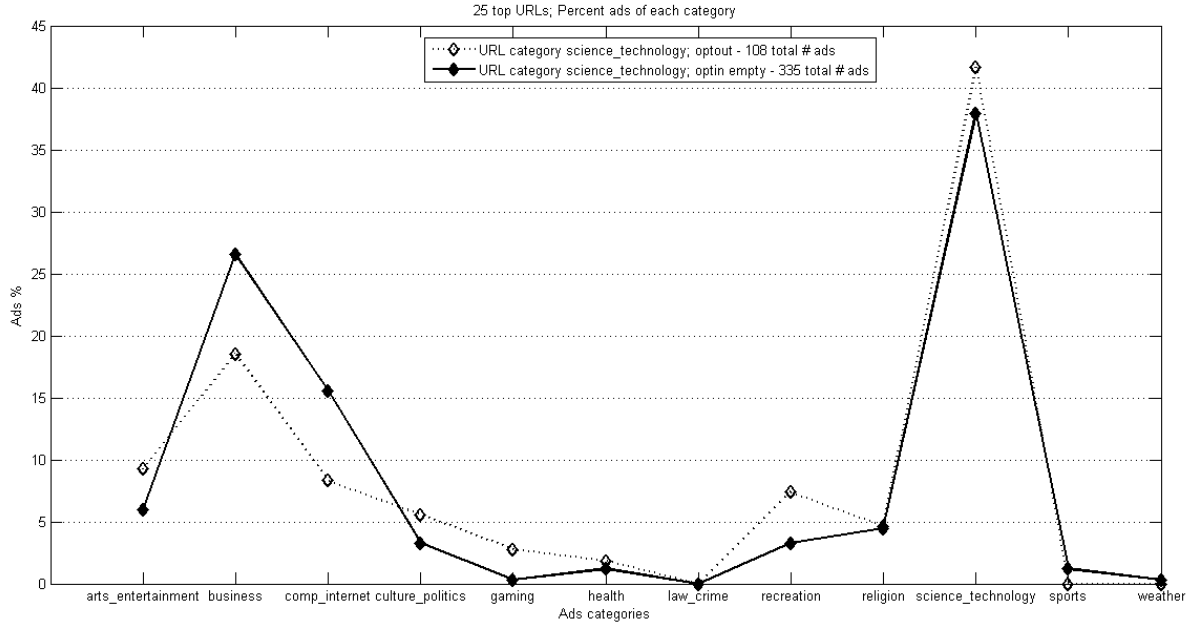


Figure 3.13: Ad baseline for URLs of category science_technology. This shows an unexpected peak in ads of type business too, which can mean that some of the URLs in this Target Categories are interpreted by Google to be of type business in fact, thus yielding contextual ads of type business for those URLs.

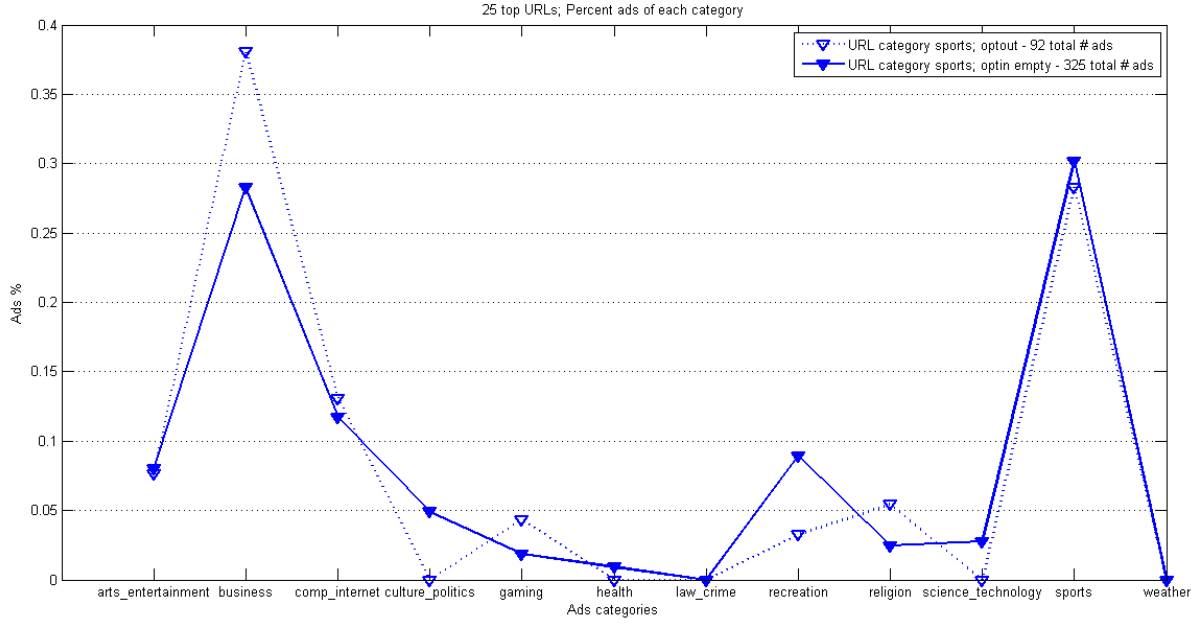


Figure 3.14: Ad baseline for URLs of category sports. This shows an unexpected peak in ads of type business too, which can mean that some of the URLs in this Target Categories are interpreted by Google to be of type business in fact, thus yielding contextual ads of type business for those URLs

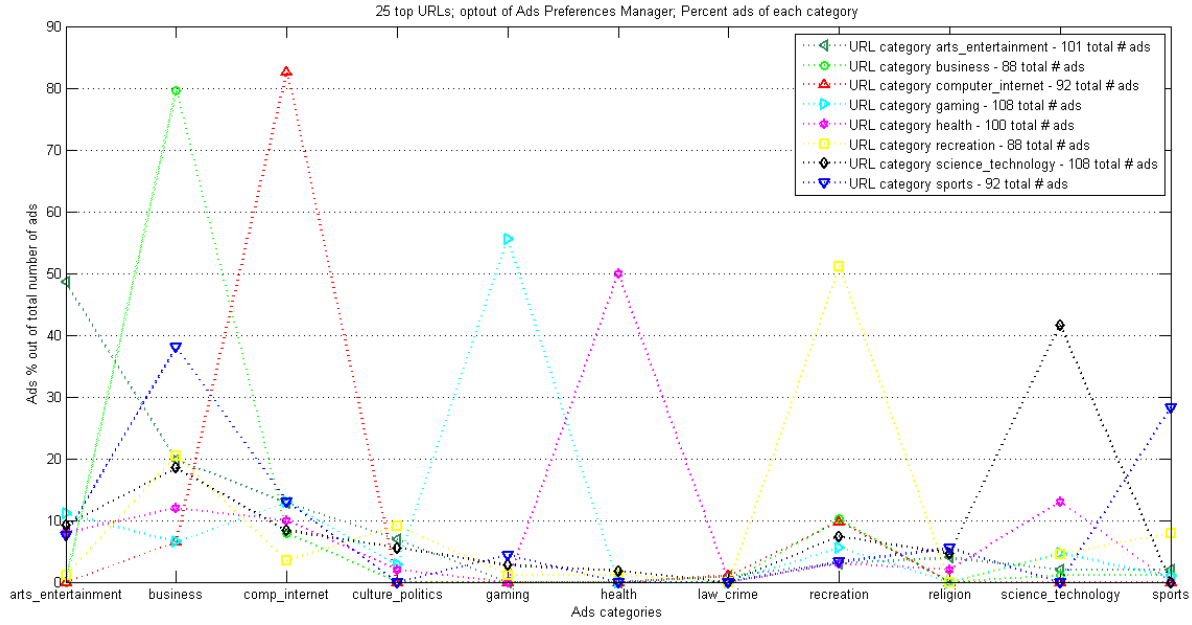


Figure 3.15: Ad baseline for target category of URLs under the optout setting.

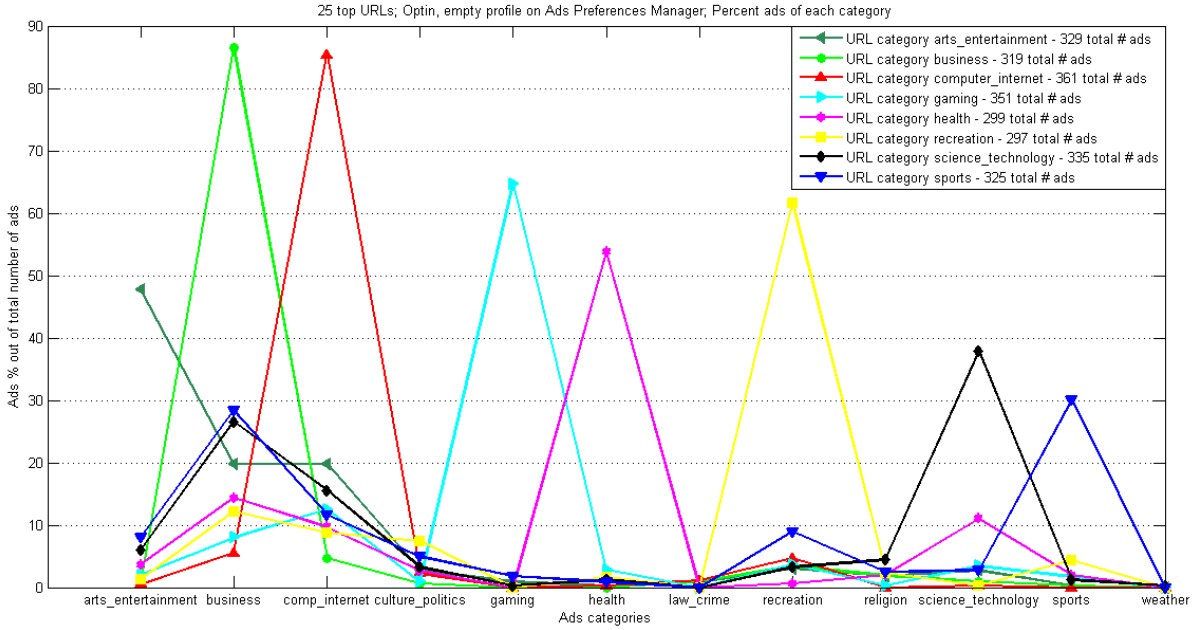


Figure 3.16: Ad baseline for target category of URLs under the optin setting, with no existing interests

Chapter 4

Results

4.1 Top1M websites with AdSense

As our first results, we described the percentage and distribution among the Alexa Top1M websites that support Google AdSense ads in section 3.2. We have produced a dataset with the individual publisher ID of each of the Top1M websites that support Google AdSense, which can be found on the EPFL server [6]. This was a time and resource consuming task and we hope the dataset can be leveraged in the future by anyone who needs to do additional AdSense studies on the top one million websites.

4.2 Websites that support OBA

Of the 200 websites from target categories we browsed in the basic experiments (25 URLs for each basic one existing interest experiment), we determine a lower bound and an upper bound of websites that delivered behavioral ads. The same setup and algorithm could be used to determine these bounds over the entire set of websites in our target categories. However, due to the fact that the sample of websites we selected was random, we believe these bounds are likely to be representative for the entire dataset.

Figures 4.1 and 4.2 show percentage of OBA supporting websites, overall and for each target category. In both lower bound and upper bound, we notice that computer_internet and business URLs are less likely to support behavioral ads that any other category websites. This could be interpreted to mean that keywords of these categories are likely to be the most expensive ones or the most efficient (generate high revenue through number of clicks) so there is no need to produce OBA to increase revenue.

4.3 Quantifying behavioral ads

We notice that the ratio number of ads / number of websites browsed from one experiment is relatively constant, regardless of the category of URLs for which ads are collected and regardless of the existing interest type. This ratio is also similar to the one obtained in the optin with empty preference profile setup - when we build the baselines for each URL target category. In our experiments, we only collect the ads in an automatic manner, so we never click the ads. It's interesting also that the number of ads we get is not reduced, even when we don't click on them.

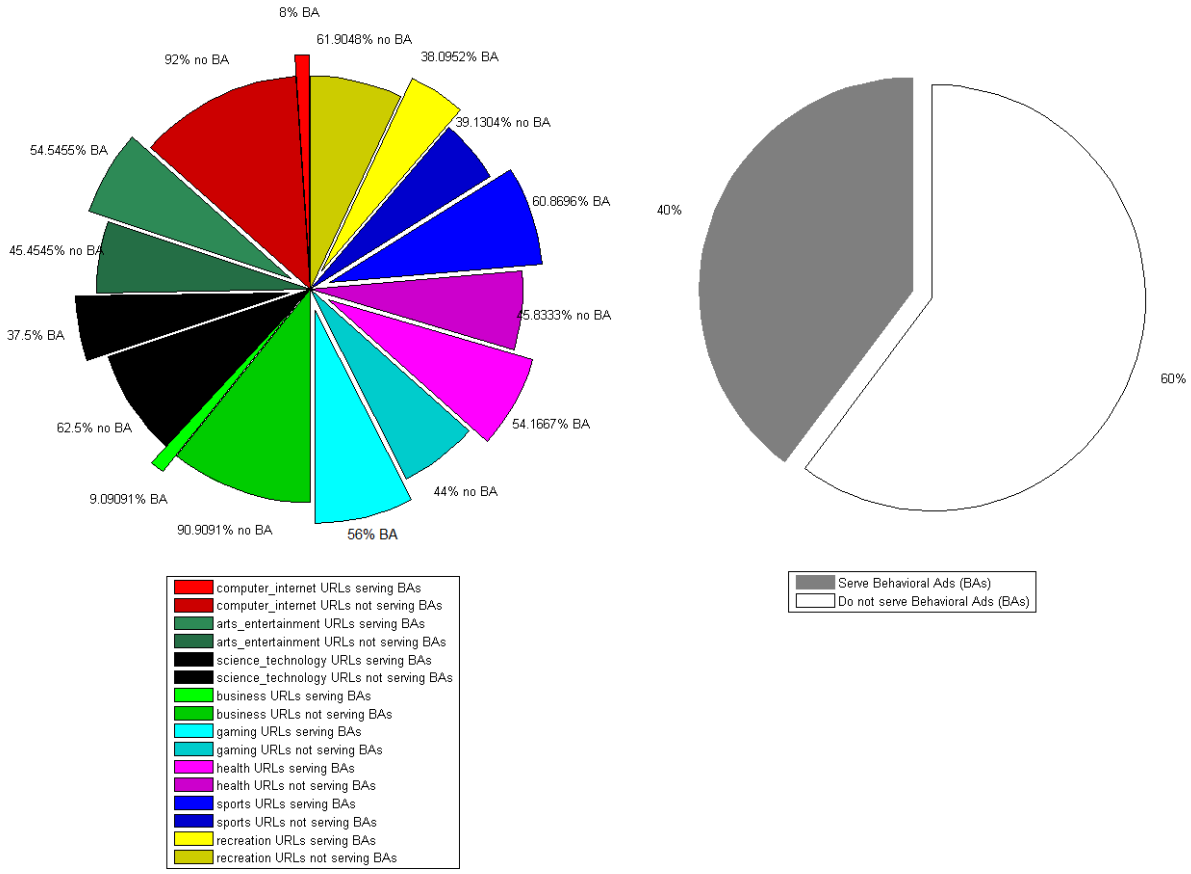


Figure 4.1: Lower bound on the number of websites from the target categories that support behavioral ads.

4.3.1 Experiments with one existing interest

We emphasize that in *all* of the experiments with one prior existing interest I, we see an increase over the baseline in number of ads of *most likely category* $= I$ for all URL categories. This means that the lower bound of behavioral ads is always positive, showing that **we receive behavioral ads, regardless of which types of URLs we browse and which existing interest we had**.

Figures 4.3, 4.4, 4.5, 4.6 and 4.7 show how the distribution of all ads categories varies in a subset of these experiments. In each, we notice the results for one URL category - C - of ads collection. We plot the distribution of ads in the baseline of category C (so with no existing interests), as well as the distribution of ads for each of the one-existing-interest-experiments for category C. By comparing the baseline with all the other experiments, we notice that there is now also *a peek in the amount of ads whose most likely category is the same as the category of the existing interest (i.e. B1 behavioral ads).* There is still a peak, but *a decrease with respect to the baseline, in contextual ads* (ads whose most likely category is the same as the URLs category C). There is little variation for ads of other categories. One exception to this last observation are experiments with arts_entertainment as prior existing interest, who also produce a peek in computer_science ads. We can explain this behavior by looking at the baseline for arts_entertainment in Figure 3.7, where we notice that URLs we classified as arts_entertainment

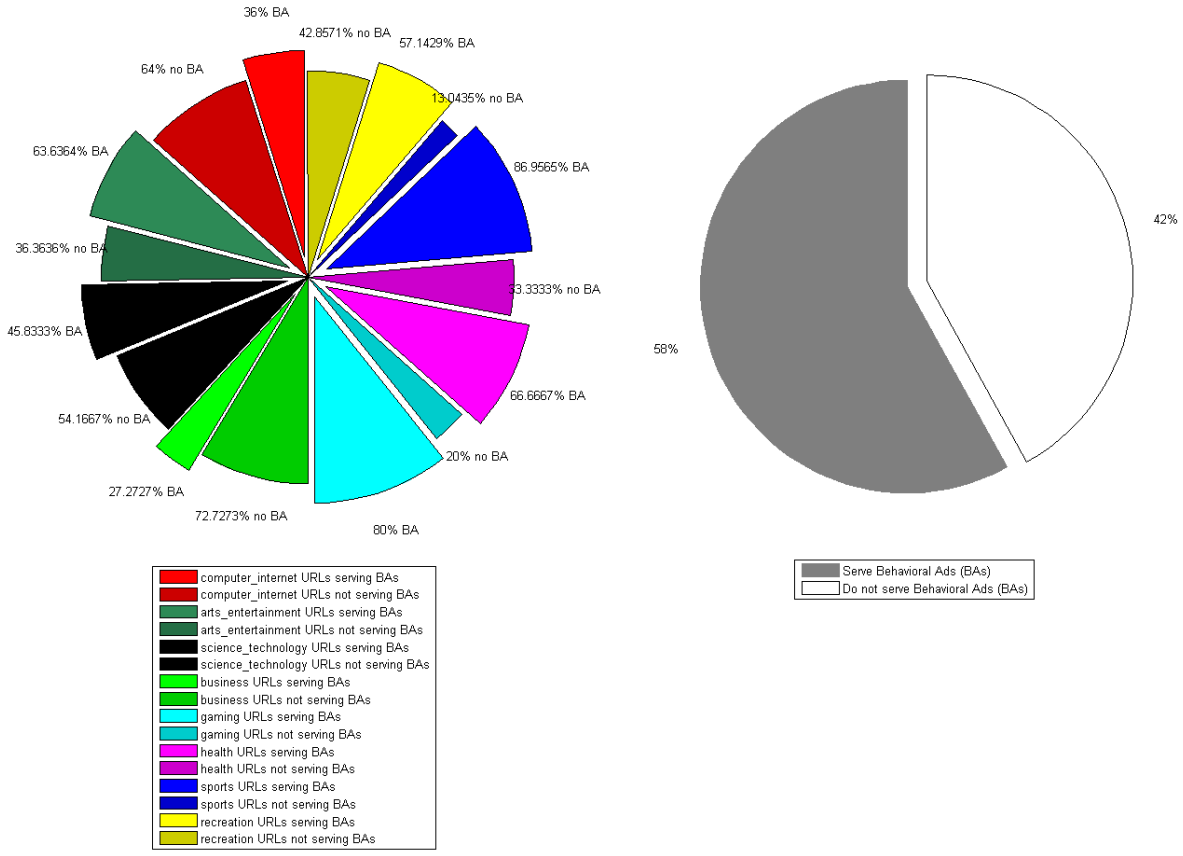


Figure 4.2: Upper bound on the number of websites from the target categories that support behavioral ads.

produce a small peak of computer_internet ads also; this can mean that, internally, Google classifies our arts_entertainment URLs as either arts_entertainment or computer_internet, thus eventually showing both these interests through behavioral ads.

As an example, Figure 4.3 emphasizes how B1 was computed. More detailed figures with the distribution of ads for several individual experiments can be seen in the Appendix A.2.

Figures 4.8 and 4.9, together with Table 4.1 show the results from all our one-existing-interest-experiments. We observe the two measures of behavioral ads, B1 and B2, support our claim that B1 would be a lower bound and B2 would be an upper bound because *B1 is always smaller than B2, regardless of existing interest and current URL category.* **On average, we measured 14.22102132 % behavioral ads with the B1 measure and 26.79652499% with the B2 measure (our of these only 4.106767728% were ambiguous, thus could also have been classified as contextual)**.

We notice that existing interest computer_internet is the one that gets showed the most via behavioral ads. The most behavioral ads are received when URLs where ads are collected is health, where the most conservative measure B1 indicates more than 50% behavioral ads.

This type of experiments can give insight to an attacker about what his fastest strategy could be in order learn end user's interest in a passive attack. For example, if the user's interest is computer_internet, than health URLs are an attacker's best chance to learn this interest because they get the most behavioral ads of this type. However, if the existing interest is gaming, than sports URLs would give the highest rate of gaming behavioral ads. So an attacker could try

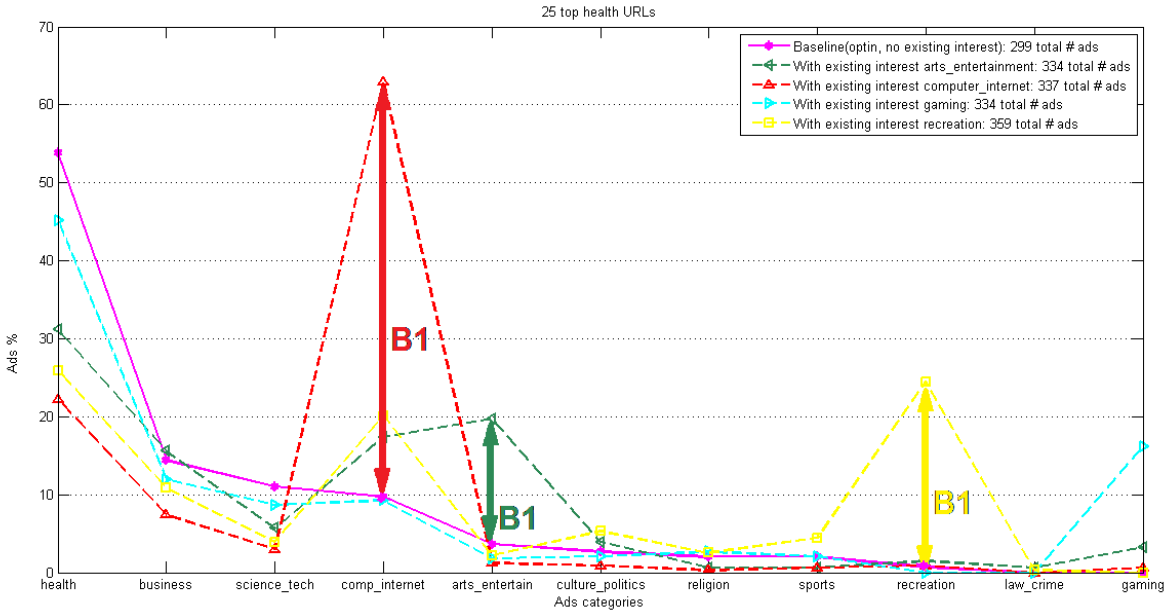


Figure 4.3: Ads distribution when browsing URLs of category health with one different existing interest, compared to the category baseline.

out all these "best chance" URL categories in order to guarantee that he learns all the user's interest.

On the other hand, these same experiments can be leveraged for an active attack. By running our experiments on different machines with a end user stolen cookie, we verified that the Ads Preferences Manager indeed gets updated with "new interests", as reflected by our external browsing. This shows that **an active attack is possible**. We can measure the success of such an attack by the amount of ads it can inflict on the user. Under the simplified assumption that an end user starts out with an empty profile, the existing interest category in the experiments can be interpreted as the category an attacker is interested in imposing to the end user; the current URLs browsed to collect ads can be interpreted as the URLs a user could become interested in, after the attack. The results of the experiments show a measure of the percentage of ads the attacked successfully imposes on the user. We can conclude that, regardless of the end user future interests, he will always receive a positive amount of behavioral ads that the attacker wanted to impose.

This scenario could be extended to measure the amount of behavioral ads an attacker can inflict when the end user profile is not empty to begin with. We think, in the worst case, the user won't see so many ads targeted by the attacker, for the simple reason that he would have its own existing interests which would contribute to part of the behavioral ads.

4.3.2 Experiments with more existing interests

Figure 4.10 and Table 4.2 show the results from our experiments with more existing interests.

On average, we now get 7.922175% behavioral ads by B1 and 46.201425% behavioral ads by B2. The lower bound measure indicates a lower amount of ads we are sure are behavioral than in the case of one-existing-interest-experiments, but the upper bound measure indicates almost twice more behavioral ads! The amount of ambiguous ads indicating how confident we are with our B2 measure is a bit higher than in the one interest experiments,

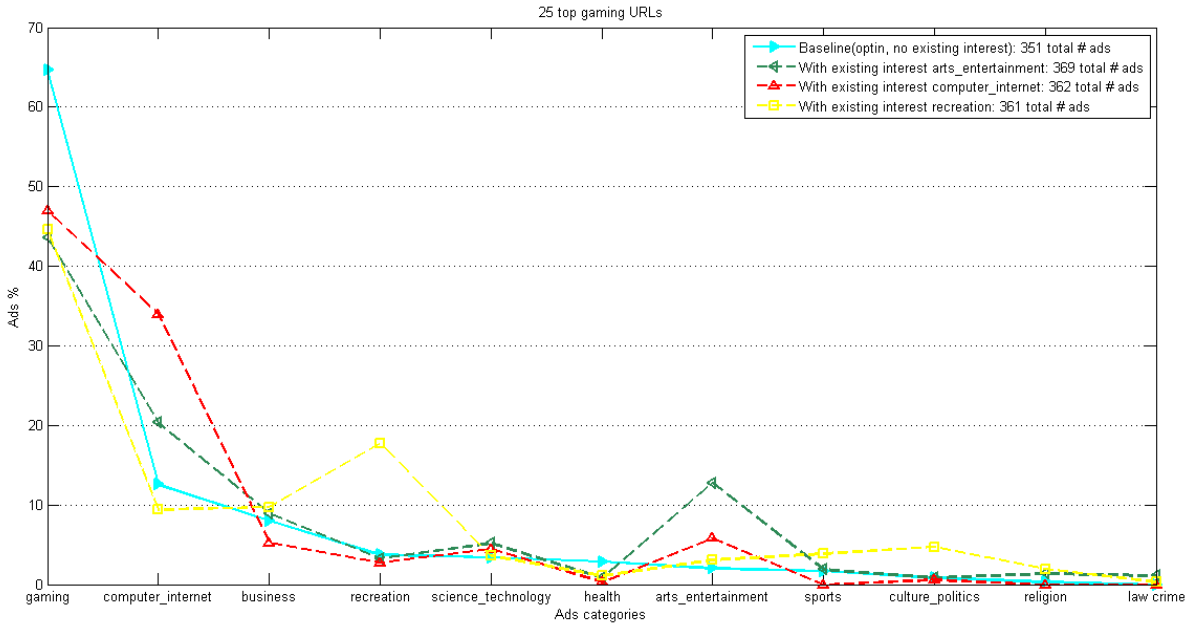


Figure 4.4: Ads distribution when browsing URLs of category gaming with one different existing interest, compared to the category baseline.

but not so high as to discount an increase of B2 behavioral ads in this multi-existing interests experiments.

The reason B1 is smaller in the multi-interests experiments is intuitive to understand. Take the example of health URLs in the 4 different experiments with one existing interest versus the one experiment with all the same 4 interests at the same time. An ad received in the multi-interests experiment can only count as B1 behavioral once (because its most likely category can only match at most one of the 4 existing interests), but it could count as B1 behavioral once for *each* of the 4 one-existing-interest-experiments.

These results indicate that **the more existing interests, the more behavioral ads are delivered**. With the B2 measure, we even get *more behavioral ads than contextual* for three out of four URL categories!

The results from these experiments, as well, can be applied to an active attack scenario. An attacker can now impose four different interests to an user with even greater success than it could impose one interest.

4.4 Ads evolution over time

In the passive attack scenario, we want to have an idea about how long it would take an attacker to learn all of the end user's interests. And how long it would take until he cannot learn any of them. In the active attack scenario, we want to know how long after an attack it takes before an end user can get ads of the type imposed by the attacker and how long until this interest no longer has influence on the end user's behavioral ads.

In order to answer these questions, we run longer one-existing-interest-experiment, which extended a few of our one-existing-interest experiments by browsing more websites. Figures 4.11, 4.14, 4.13 and 4.12 show our results. These illustrate the fact that **the amount of behavioral ads (as measured by both B1 and B2) drops with the number of websites**

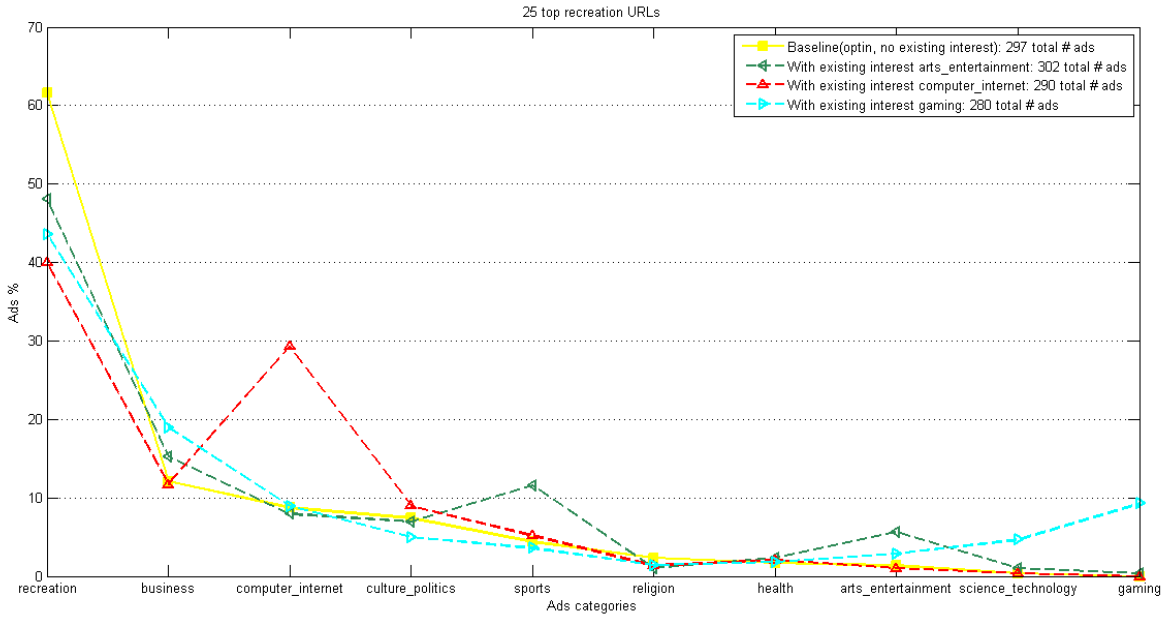


Figure 4.5: Ads distribution when browsing URLs of category recreation with one different existing interest, compared to the category baseline.

browsed (so with time too). In this case, **with only one existing interest**, this is revealed with the first or second website browsed, thus rather **quickly**. Moreover, 200 websites are not enough to fully erase one existing interest from the behavioral ads (i.e. *one existing interest is not lost*). Specifically, for an active attack this means the attacker is guaranteed that the user will not only get the ads they wanted to impose *immediately after the attack*, but that these will last for at least the next 200 websites the end user visits. For a passive attack, this says that *one or two websites are enough for an attacker to surely learn the end users' interest.*

We also notice a **decrease with time in behavioral ads as measured by both B1 and B2 when looking at the experiments with four existing interests**. Figures 4.15, 4.16, 4.17 and 4.18 show the evolution of B1 and B2 behavioral ads, as well as Contextual ads. We notice that most of the time, **all the interests are revealed quickly through behavioral ads**, among the first few browsed websites. One exception is the arts_entertainment interest, which is revealed *surely* only with the 40th website for health websites, and *never* for science_technology websites. We mention here than the websites to collect ads for are selected from the target categories without taking into account if they deliver behavioral ads. As we've described in 3.9, we have identified for each website, if it delivers behavioral ads. In a future setup, we could take advantage of this information and only browse websites that were identified to serve behavioral ads in order to determine how soon interests can be revealed. We believe in this case, interests could be revealed sooner.

There is some variation regarding the time when none of the interests show through behavioral ads in this multi-existing-interests-experiments. For business and health URLs, none of the four interests is lost after the 200 websites browsed. None of the three interests that are revealed with science_technology URLs disappear either. With sports URLs, two of the interests disappear rather quickly - recreation and arts_entertainment. However, in all of these experiments there are at least two existing interests still revealed through B1 behavioral ads after browsing 200 websites. Which means that, as in the case of one existing interest only,

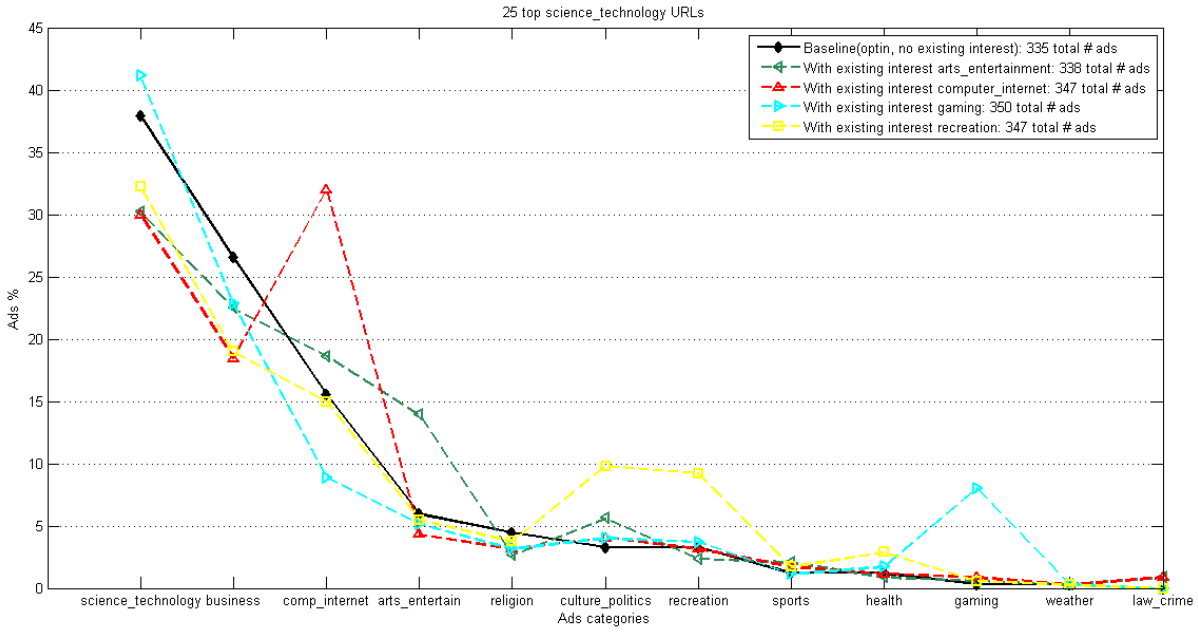


Figure 4.6: Ads distribution when browsing URLs of category science_technology with one different existing interest, compared to the category baseline.

behavioral ads are still surely present after 200 websites.

Therefore, when browsing many websites **immediately** *after the interest profile was bulit*, we saw an overall decrease of behavioral ads, but behavioral ads do no completely disappear. However, in the time we let pass between two experiments we did notice that from the Ads Preference Manager, interests are removed rather rapidly. Typically, from one evening when we would finish an experiment, to the next morning, the interests would disappear from the Ads Preference Manager. This gave the idea of a new kind of experiment, which would also show the evolution of ads after some time. We now setup a one-interest preference profile, we run a basic one-existing-interest experiment - meaning we collect the ads immediately after the profile was built. After 12 hours we re-run the same basic experiment without rebuilding the preference profile. According with our empirical observation of the Ads Preferences Manager being cleared out, we notice that the decline in behavioral ads is noticeable. Figure 4.19 shows that after 12 hours, our B1 measure indicates no behavioral ads and, consequently, there is a definite increases in contextual ads.

This observation shows that only immediate (the most recent) interests of an end user can be determined by our usual approach - in most of our experiments we build a user preference profile rapidly, browsing websites one after another. In real life, users typically take more time in building their preference profiles, so an interesting future direction of study would be to build different kind of user profiles, according to different activity rates (slowly built profiles, rapidly built profiles). We notice empirically that with fewer interests, these are kept longer in the Ads Preferences Manager, thus can produce behavioral ads for a longer period of time than a rapidly built profile can. So it is likely that older interests could be found in this case.

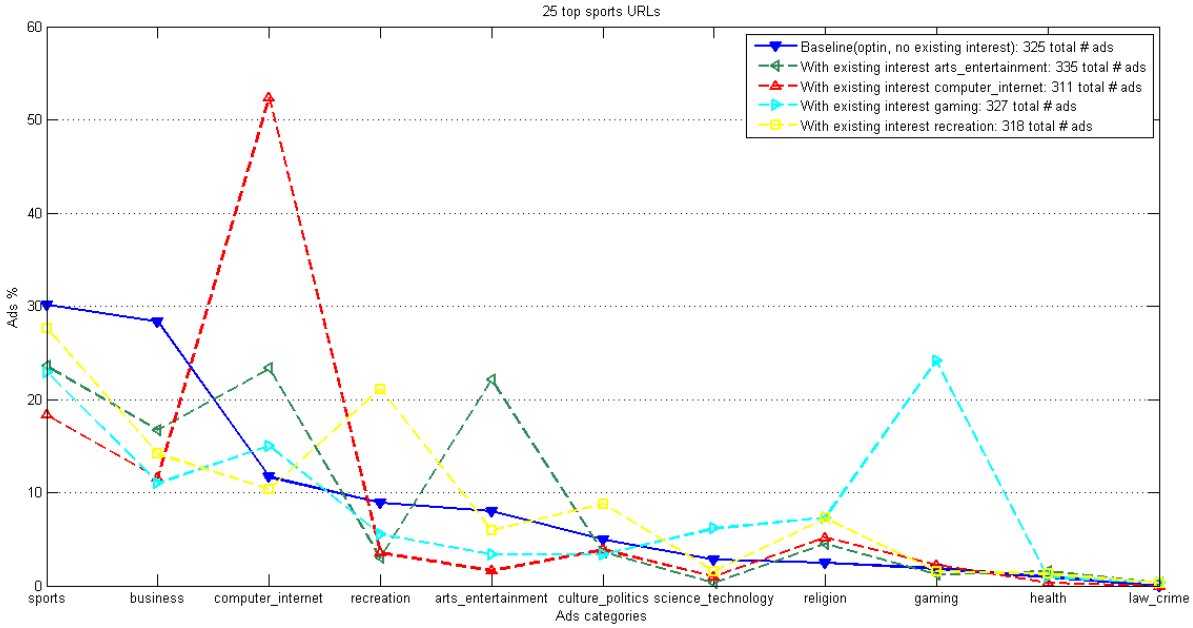


Figure 4.7: Ads distribution when browsing URLs of category sports with one different existing interest, compared to the category baseline.

Table 4.1: Behavioral ads measures for experiments with one existing interest. Ambiguous ads are those that we could be either behavioral by the B2 measure, or contextual. URLs category refers to the category of URLs for which ads were collected in one experiment.

Existing interest	*URLs category*	*B*1 (%)	*B*2 (%)	*Ambiguous* (%)
arts_entertainment	computer_internet	5.050703144	8.554572271	2.359882006
	science_technology	7.93517619	15.97633136	0.591715976
	business	2.695299889	11.29568106	6.644518272
	recreation	4.282337726	8.609271523	1.655629139
	gaming	10.74282538	17.88617886	2.7100271
	health	16.08154928	22.15568862	0.8982
	sports	14.08955224	25.67164179	0.59701
computer_internet	arts_entertainment	26.76884085	60.72507553	6.344410876
	science_technology	16.46608456	44.95677233	5.187319885
	business	12.42109331	31.16438356	13.69863014
	recreation	20.55613607	43.10344828	8.965517241
	gaming	21.44228802	55.24861878	15.74585635
	health	53.20901521	69.43620178	2.077151335
	sports	40.71926787	62.37942122	3.215434084
gaming	computer_internet	1.724137931	6.896551724	4.885057471
	arts_entertainment	8.822659171	13.86430678	1.179941003
	science_technology	7.701492537	9.428571429	0.571428571
	business	4.262295082	4.918032787	0
	recreation	9.285714286	16.07142857	3.571428571
	health	16.16766467	18.56287425	0.299401198
	sports	22.31286756	29.05198777	0.611620795
recreation	computer_internet	0.846414281	9.356725146	2.631578947
	arts_entertainment	11.28975461	25.6097561	6.097560976
	science_technology	5.938319928	17.29106628	1.729106628
	business	7.362534949	19.93243243	6.418918919
	gaming	14.02482815	30.19390582	9.141274238
	health	23.8436385	34.54038997	0.557103064
	sports	12.14610547	37.42138365	6.603773585
Avg for any existing interest and URLs categ.		*14.22102132*	*26.79652499*	*4.106767728*

Table 4.2: Behavioral ads measures for experiments with more existing interests at a time. Existing interests in each experiment: *arts_entertainment*, *computer_internet*, *gaming and recreation*.

Current URLs category		**Health**	**Sports**	**Sci_tech**	**Business**	*Average*
*B*1(%) *of type*	*arts_entertainment*	1.4637	0	0	0.7047	*0.5421*
	computer_internet	4.2595	8.0751	5.8121	3.803	*5.487425*
	gaming	1.0165	0.7563	1.274	0.2148	*0.8154*
	recreation	2.3336	0	1.9013	0.0741	*1.07725*
*B*1(%) *behavioral ads* (*of any type*)		9.0733	8.8314	8.9874	4.7966	*7.922175*
*B*2(%) *behavioral ads*		43.2152	50.8306	52.4437	38.3162	*46.201425*
Contextual ads(%)		37.8133	40.3101	31.0242	79.6821	*47.207425*
Ambiguous ads(%)		8.9023	9.9668	10.9222	23.11	*13.225325*

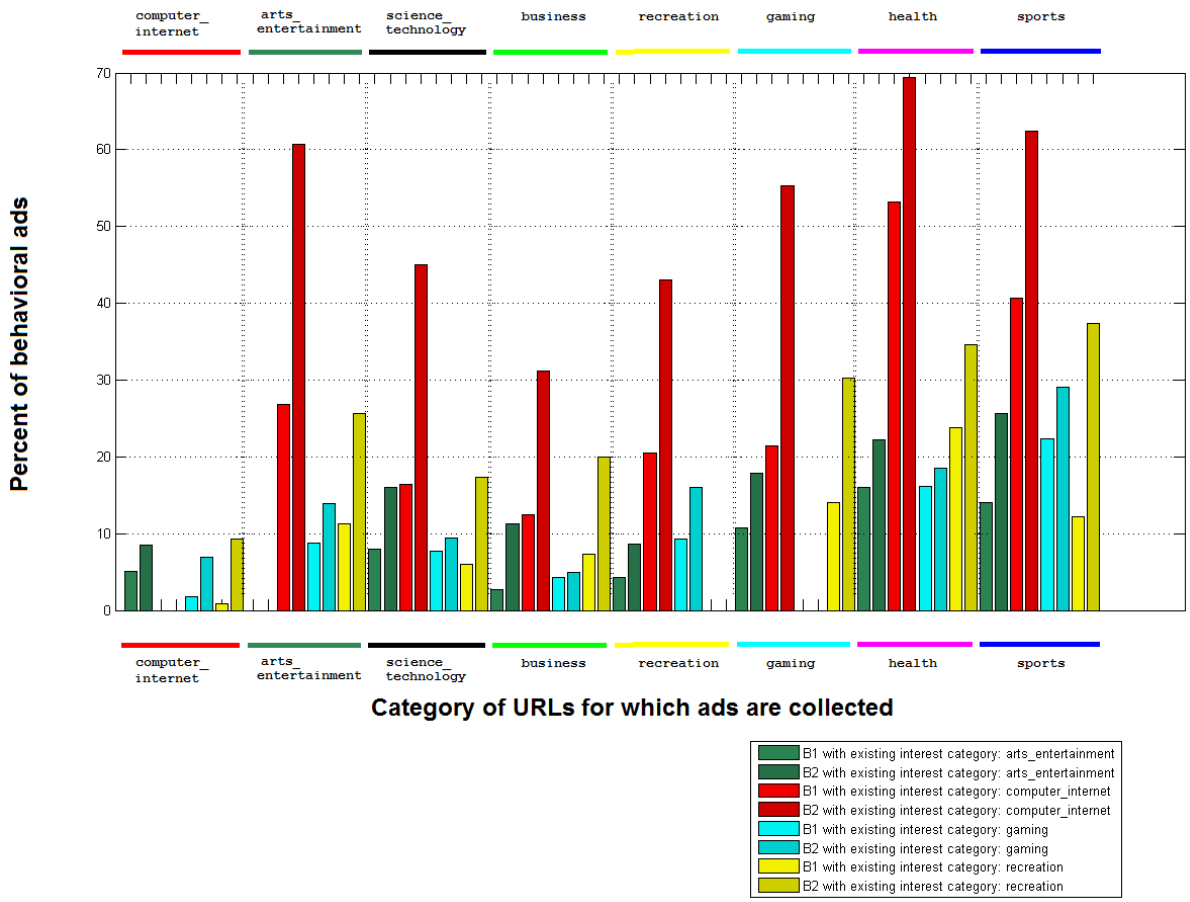


Figure 4.8: B1 and B2 measure for behavioral ads in all combinations of existing interest and URL category for which ads were retrieved.

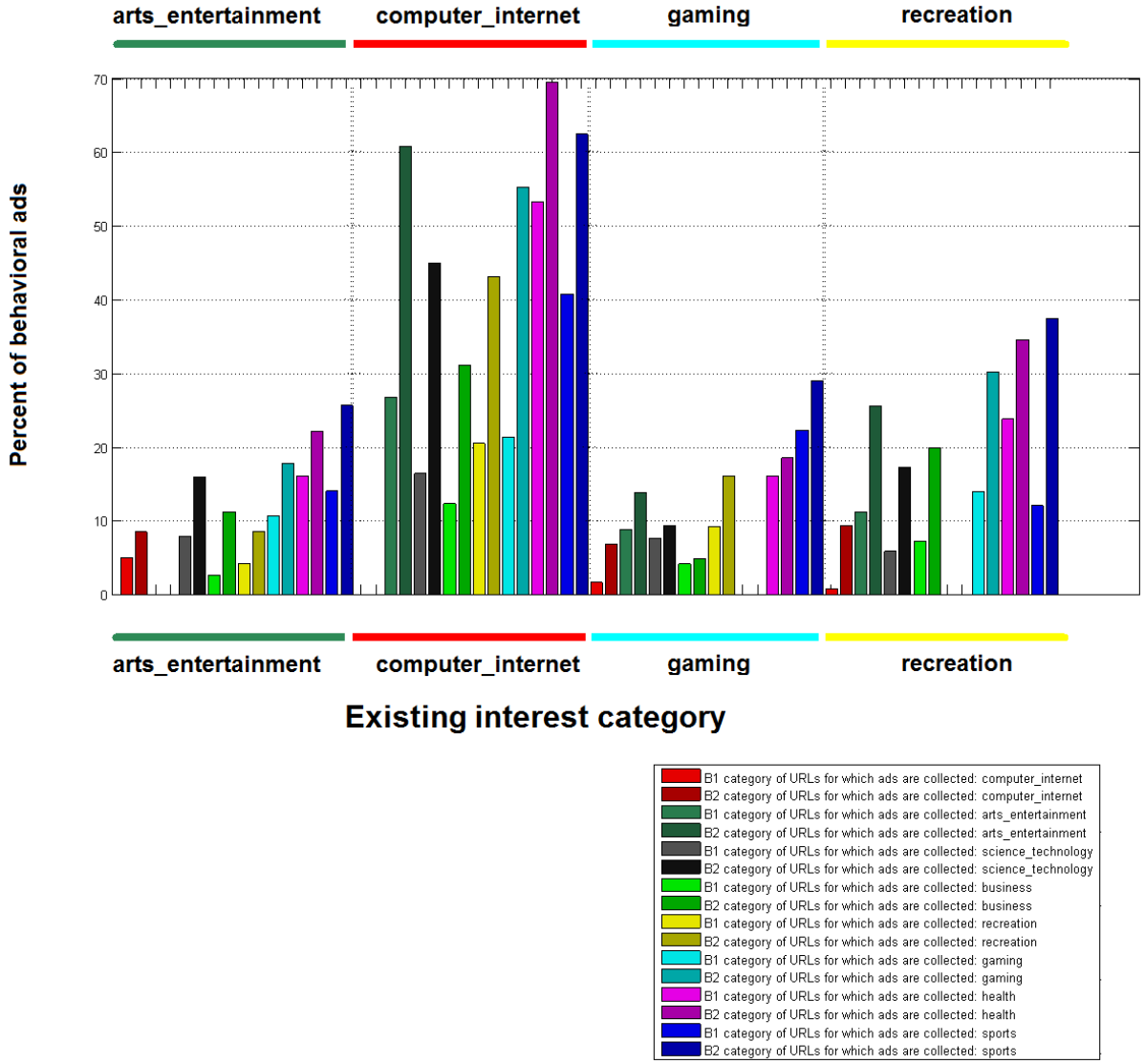


Figure 4.9: B1 and B2 measure for behavioral ads in all combinations of existing interest and URL category for which ads were retrieved.

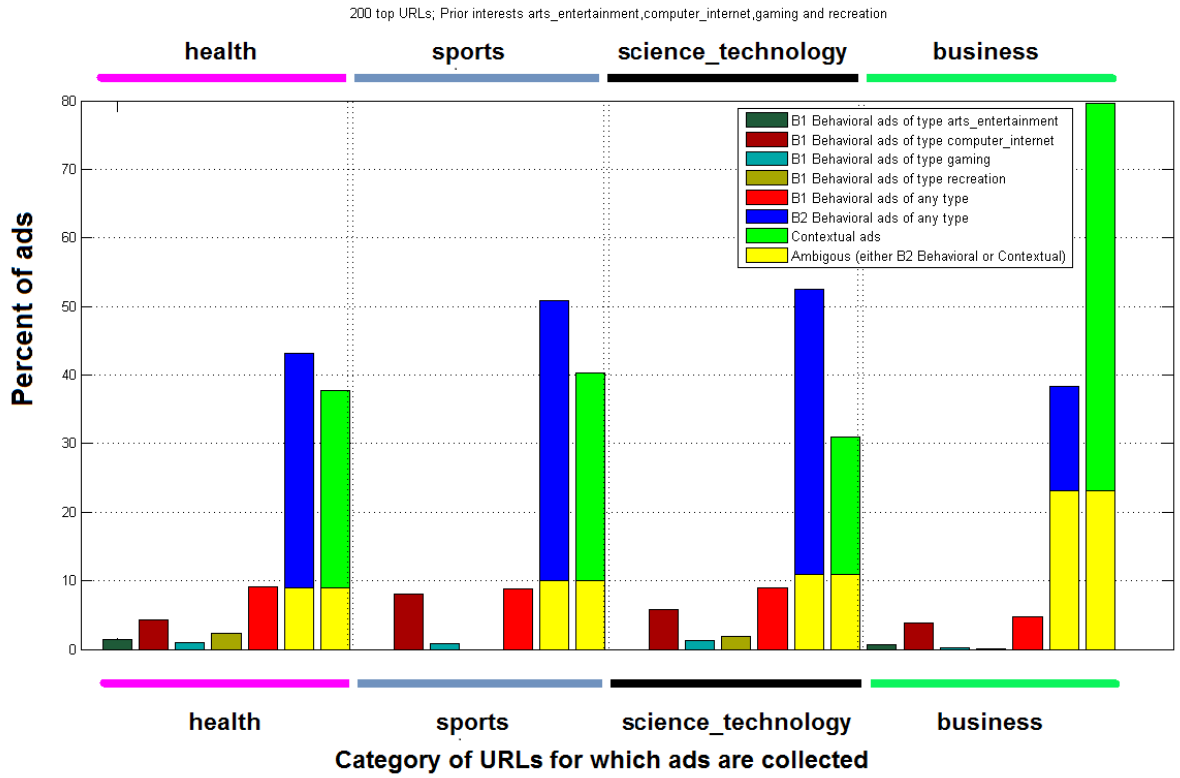


Figure 4.10: B1, B2 behavioral ad measures, contextual and ambiguous ads after 200 websites were browsed, with 4 existing interests: arts_entertainment, computer_internet, gaming and recreation. Category of URLs for which ads were retrieved: business, health, sports and science_technology.

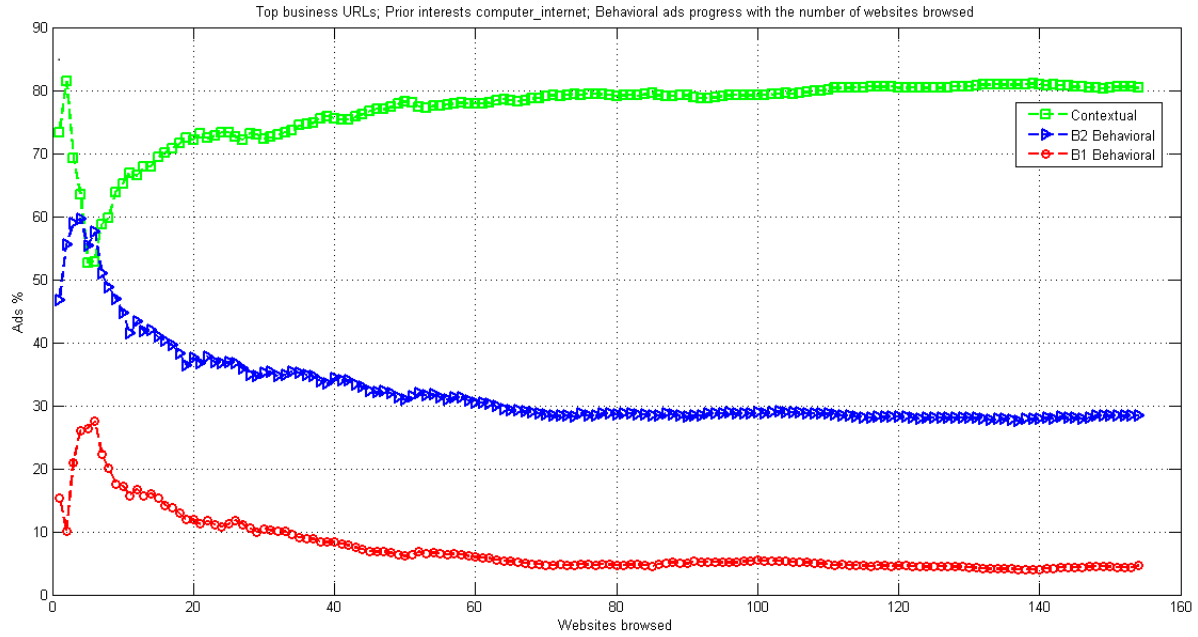


Figure 4.11: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **business** with existing interest computer_internet. Behavioral ads are still present after nearly browsing nearly 160 websites.

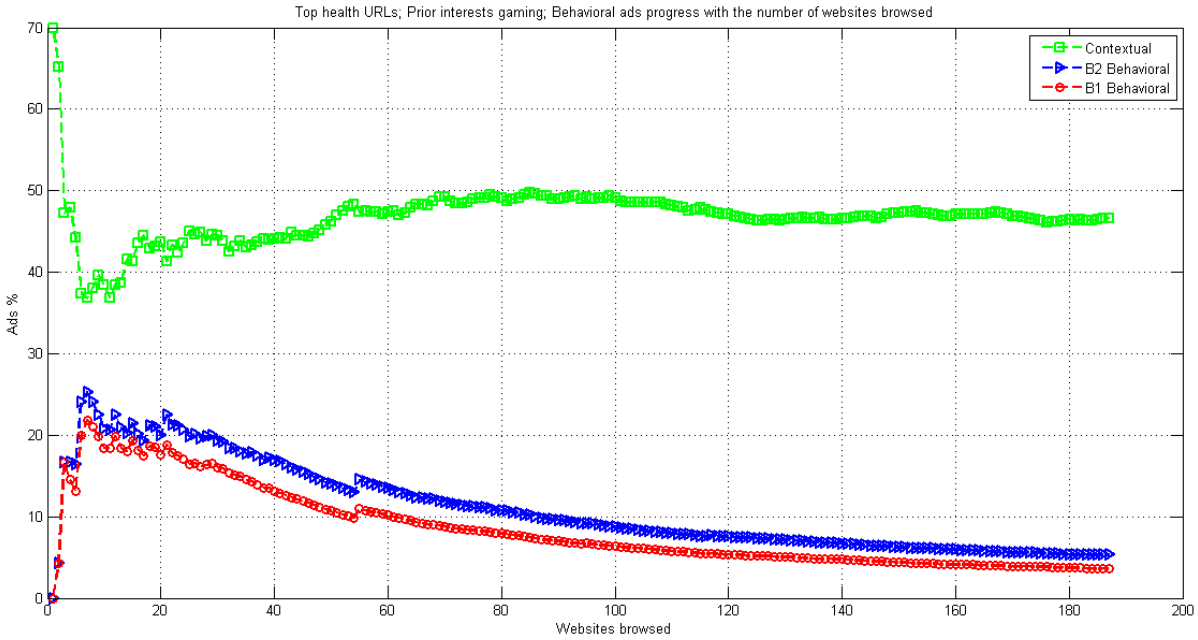


Figure 4.12: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category health with existing interest gaming. Behavioral ads are still present after nearly browsing nearly 200 websites.

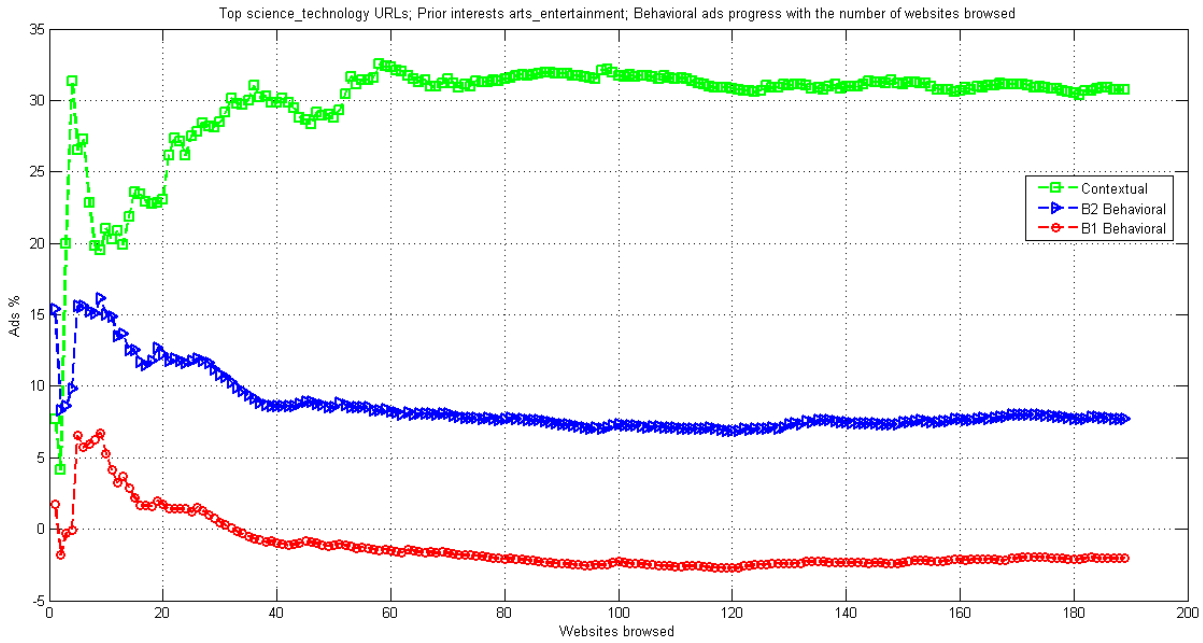


Figure 4.13: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category science_technology with existing interest arts_entertainmnet. B2 indicates behavioral ads are still present after nearly browsing 200 websites, but B1 indicates behavioral ads are gone little after 30 websites..

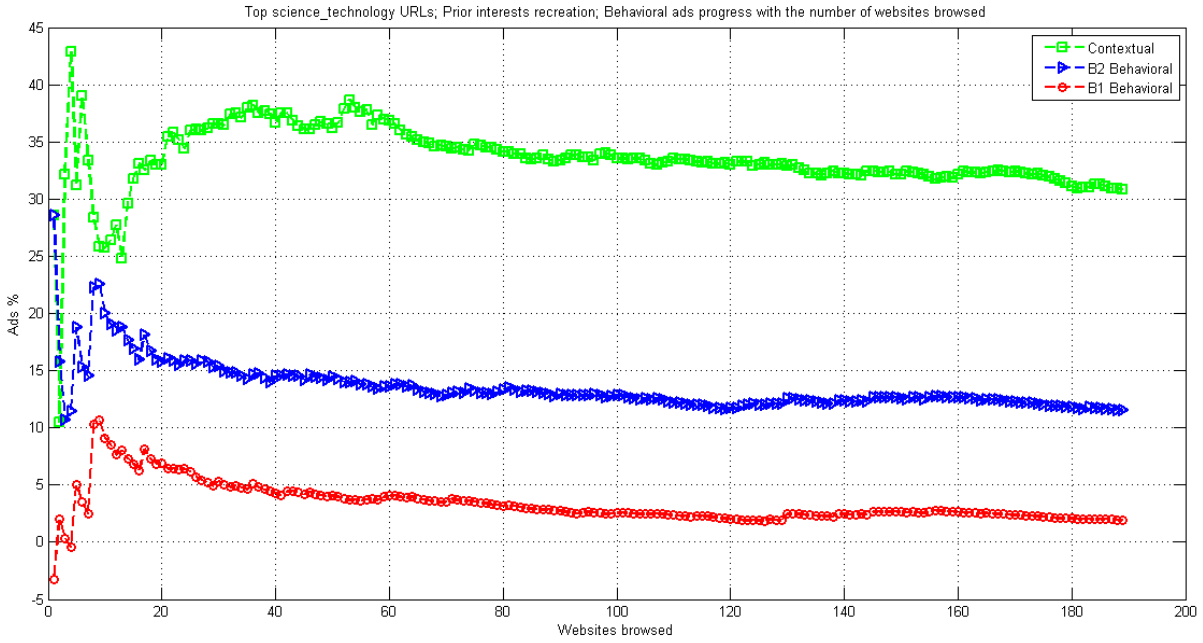


Figure 4.14: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category science_technology with existing interest recreation. Both B1 and B2 indicate that behavioral ads are still present after browsing 200 websites.

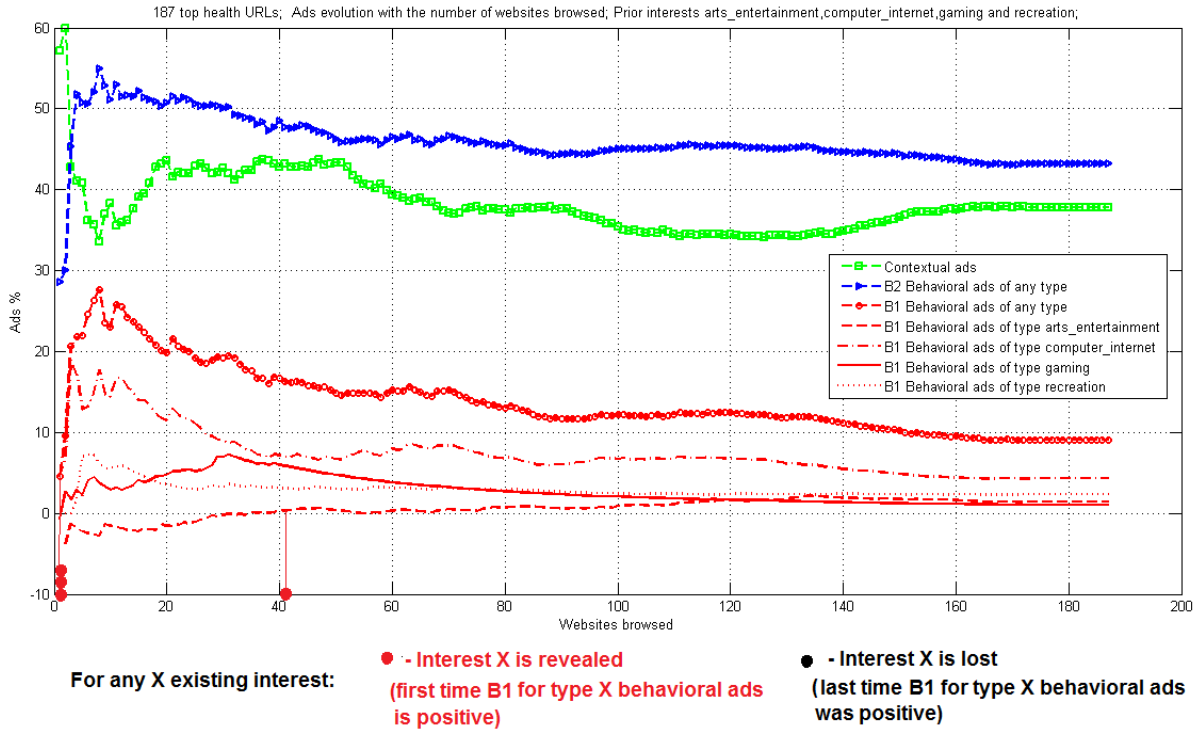


Figure 4.15: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **health** with 4 existing interests: arts_entertainment, computer_internet, gaming and recreation. Notice that interests computer_internet, gaming and recreation are revealed with the first website (red dot), while arts_entertainment is revealed after approximately 40 websites (red dots). In the 187 websites we visit, none of these interests are lost (no black dots)

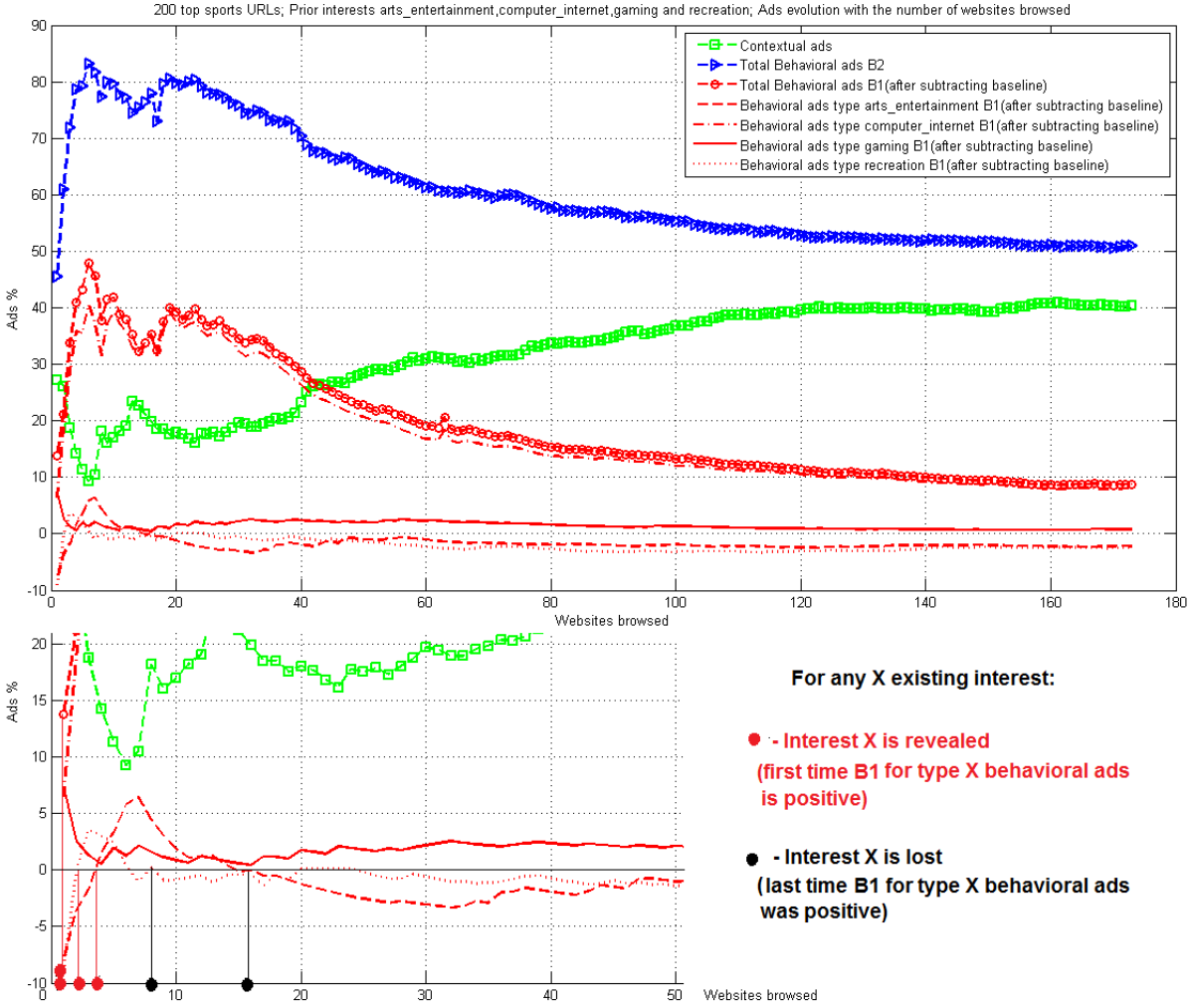


Figure 4.16: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **sports** with 4 existing interests: arts_entertainment, computer_internet, gaming and recreation. Notice that all interests are revealed in the first 5 websites, but the interest in recreation is lost very quickly, after 8 websites (black dot), as well as the interest in arts_entertainment, after17 websites (black dot). The other two interests were not lost in the 200 websites we browsed.

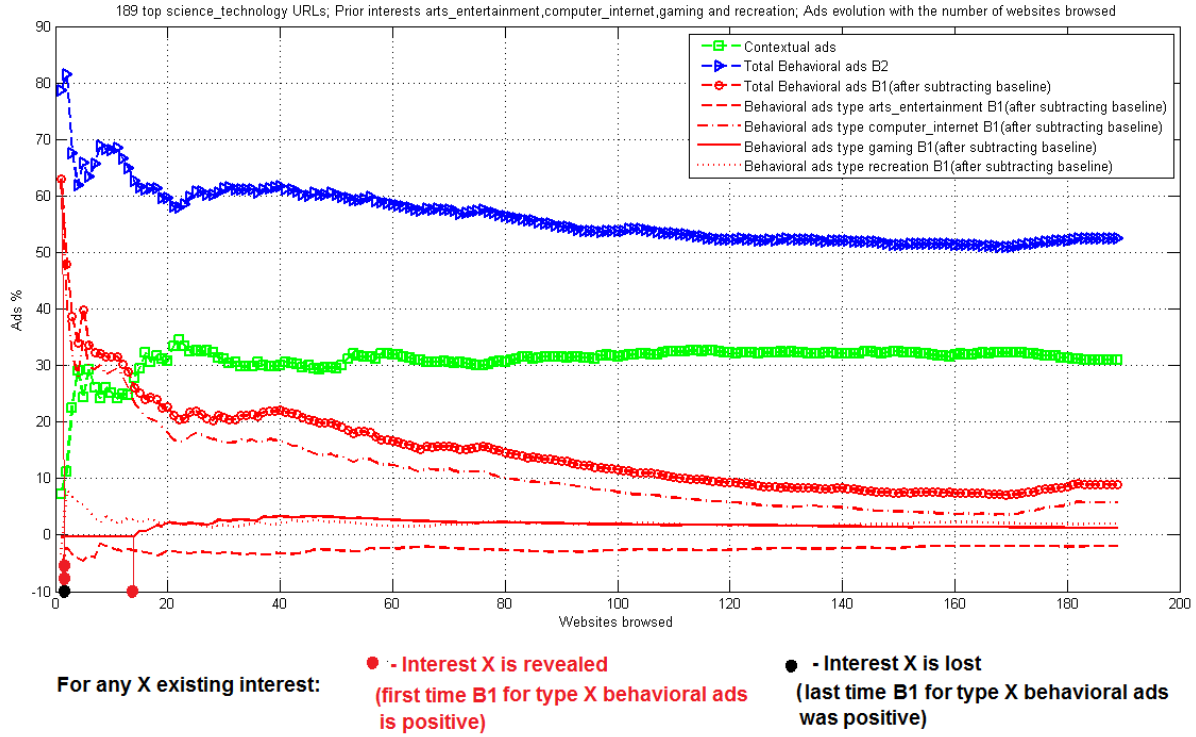


Figure 4.17: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **science_technology** with 4 existing interests: arts_entertainment, computer_internet, gaming and recreation. Notice that the interests computer_internet and recreation are revealed from the first websites (red dots), the interest in gaming is revealed after 14 websites (red dot), but the interest in arts_entertainment is never revealed in all the 189 websites that we browse (black dot). The other three interests are not lost in this time.

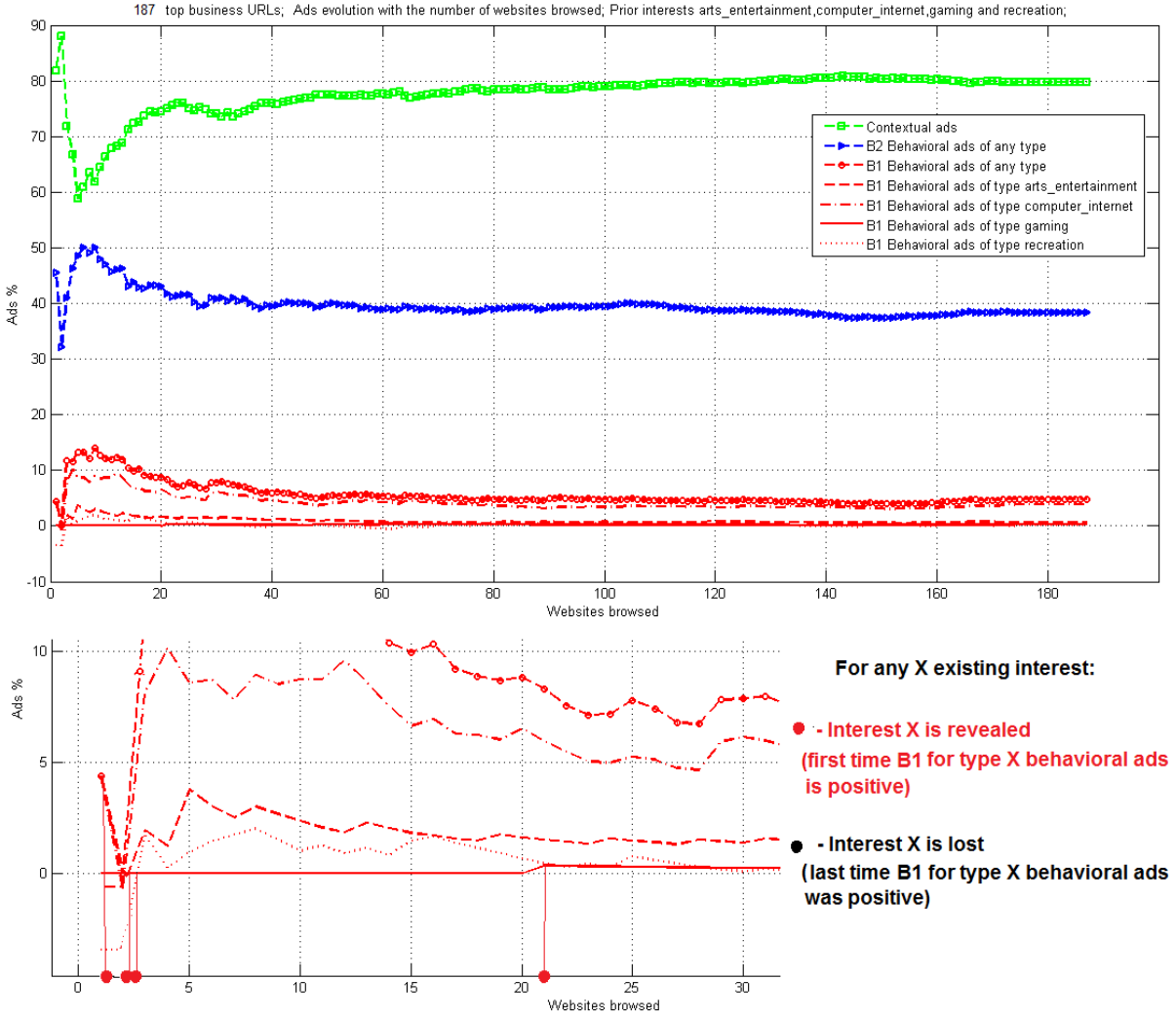


Figure 4.18: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **business** with 4 existing interests: arts_entertainment, computer_internet, gaming and recreation. Notice that interest computer_internet is the first one revealed (with the first website - red dot), closely followed by arts_entertainment and recreation in the second website (red dots), while gaming is only revealed after 21 websites (red dot). None of them are lost in the 187 URLs that we browse.

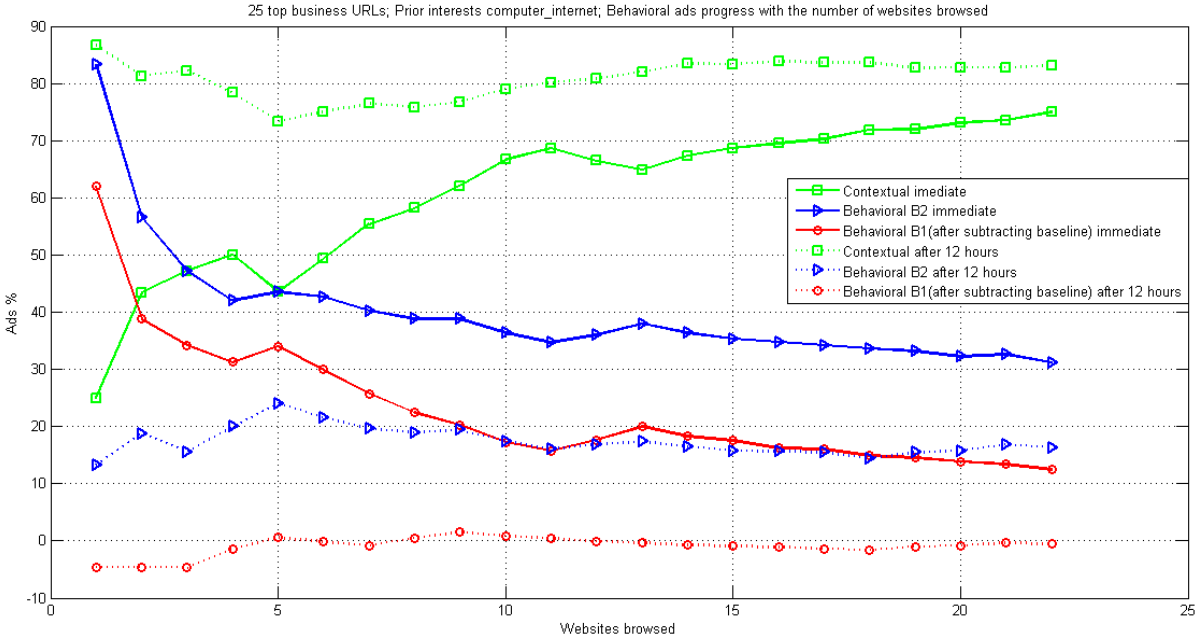


Figure 4.19: Evolution with respect to the number of websites browsed of B1, B2 behavioral ad measures and contextual ads for URLs of category **business** with existing interest computer_internet in two different settings: 1) immediately after the preference profile was built and 2) 12 hours after.

Chapter 5

Conclusions and Future Work

In our work, we firstly identified general patterns to detect if websites support Google AdSense ads. We produced a dataset of the top 1000000 websites supporting such advertising (as ranked by Alexa.com) and we analyzed their distribution. We observed an average of 14.22% websites with Google AdSense; the first few thousands of very popular websites support Google AdSense advertising in a small percent, after which a peak is quickly reached; beyond the top 300.000th website there is a small, constant decrease. This evolution could be explained by the fact that popular websites deal with advertisers themselves and Ad sense is better suited for the "heavy tail" websites, as it allows cheap, automatic ad inclusion. The gradual decrease also suggests that, from some rank on, websites that are not too popular are less likely to support ads. This can be explained by the fact that a website which is not popular will not have enough visitors to generate revenue from clicking on ads, thus its not rational for it to join the AdSense program. A larger dataset of ranked websites would be needed to further study this intuition.

Secondly, we have understood how Google AdSense works and reverse-engineered the ad collection process. From our reduced dataset of websites that support Google AdSense which we used in experiments, we determined that between 40% and 58% websites deliver behavioral ads. Our algorithms can be applied to determine the websites that deliver behavioral ads for the entire dataset. Future work could also analyze the distribution with respect to the global rank of websites that deliver behavioral ads, as well as look at how many behavoral ads are delivered on individual websites.

Additionaly, we defined an algorithm to classify ads as contextual, as well as two measures to quantify the amount of behavioral ads - a lower bound measure and an upper bound measure. We have seen that we always receive behavioral ads, *regardless* of what our existing or present interests are. We looked at both one existing interest and multi existing interests scenarios and we noticed that more behavioral ads are received when having more existing interests.

Surprisingly, we showed that an *active attack is possible*: one or more interests can be imposed on an end user and behavioral ads of those types are received by the user *immediately.*

Furthermore, we showed that, for a passive attack, end user's interests can be learned *quickly* (one existing interest is revealed with certainty in the first or second browsed website after the interest was established). We've seen that typically, the interests are not lost after browsing 200 websites, even though there is a decrease with time of behavioral ads. For an active attack, this can mean that an end user is guaranteed to see the effect of the attack through ads on at least the next 200 websites that he will browse. Future work can be done to study the duration of this effect, by increasing the number of websites browsed.

Finally, we noticed that only immediate interests can be learned - after 12 hours end users interests are untraceable. However, our scenario artificially builds end user profiles in a rapid fashion, by browsing websites of several categories one after another - this would correspond

to a very active user. We noticed that if profiles are built slower, interests can persist for a longer period of time, thus future work could study the evolution over time of ads distribution by taking into account end user profiles built at different speeds.

Future work could also include adding other categorization tools in order to improve our current categorization, retrieving image and video ads (which could be converted to text by OCR or video processing techniques and categorized as text ads, or directly classified through machine learning techniques), and studying the use of aggregated interests; we could try to determine if Google uses inferences made from aggregated user information about end user interests in order to deliver ads. For instance, they could infer that end users interested in sports and likely to be interested in cars too, so that when a particular user becomes interested in sports, they could show him car ads by association.

Appendix A

A.1 Determining if a website supports Google AdSense ads

A.1.1 Pattern 1

An example can be seen in the source page of www.beethoven.com

```
google_ad_client = "pub-7882126374182169";
```

The general code to look for such a pattern is:

```
String searchRegEx =
 ".*(google_ad_client)[ ]*[=]{1}[ ]*[\"\’][ ]*([a-zA-Z-\\d]+)[\"\’].*"
if (source.matches(searchRegEx)) pubID = source.replaceAll(searchRegEx, "$2"),
```

where `source` is the source of the website, as a String.

A.1.2 Pattern 2

An example can be seen in the source page of www.gayot.com

```
GS_googleAddAdSenseService("ca-pub-0358308655580959")
```

The general code to look for such a pattern is:

```
String searchRegEx = "".*(GS_googleAddAdSenseService)[ ]*[(]{1}[ ]*[\"\’][ ]*
([a-zA-Z-\\d]+)[\"\’][ ]*[)]{1}[ ]*.*";"
if (source.matches(searchRegEx)) pubID = source.replaceAll(searchRegEx, "$2"),
```

where `source` is the source of the website, as a String.

A.1.3 Pattern 3

An example can be seen in the source page of www.oprah.com

```
..., "google_ad_client":"ca-owntv", ...
```

The general code to look for such a pattern is:

```
String searchRegEx =
".*[\’\"]{1}(google_ad_client)[ \’ \"]{1}[ ]*[:]{1}[ ]*
[ \" \’][ ]*([a-zA-Z- \ \d]+)[ \" \’].*";
if (source.matches(searchRegEx)) pubID = source.replaceAll(searchRegEx, "$2"),
```

where `source` is the source of the website, as a String.

A.1.4 Pattern 4

An example of this pattern is to find in the source page of a website:

```
GA_googleFillSlotWithSize("ca-pub-0123456789",....)
```

The general code to look for such a pattern is:

```
String searchRegEx =
 ".*(GA_googleFillSlotWithSize)[ ]*[(]{1}[ ]*[\"\']?[ ]*
([a-zA-Z-\\d]+)[\"\'].*[)]{1}.*";
if (source.matches(searchRegEx)) pubID = source.replaceAll(searchRegEx, "$2");
```

where `source` is the source of the website, as a String.

A.2 Experimental results with one existing interest category

We additionaly show a few plots for the experiments with one existing interests and 25 websites for which ads are collected in Figures A.1, A.2, A.3, A.4, A.5 and A.6.

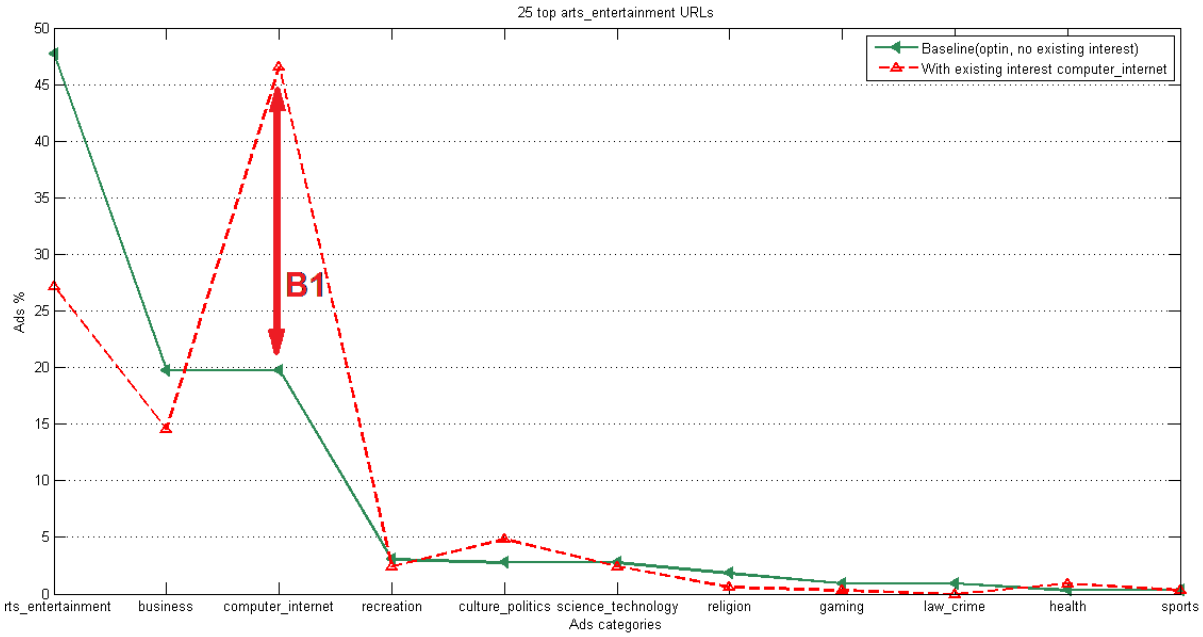


Figure A.1: Ads distribution when browsing URLs of category arts_entertainment with existing interest computer_internet, compared to the category baseline.

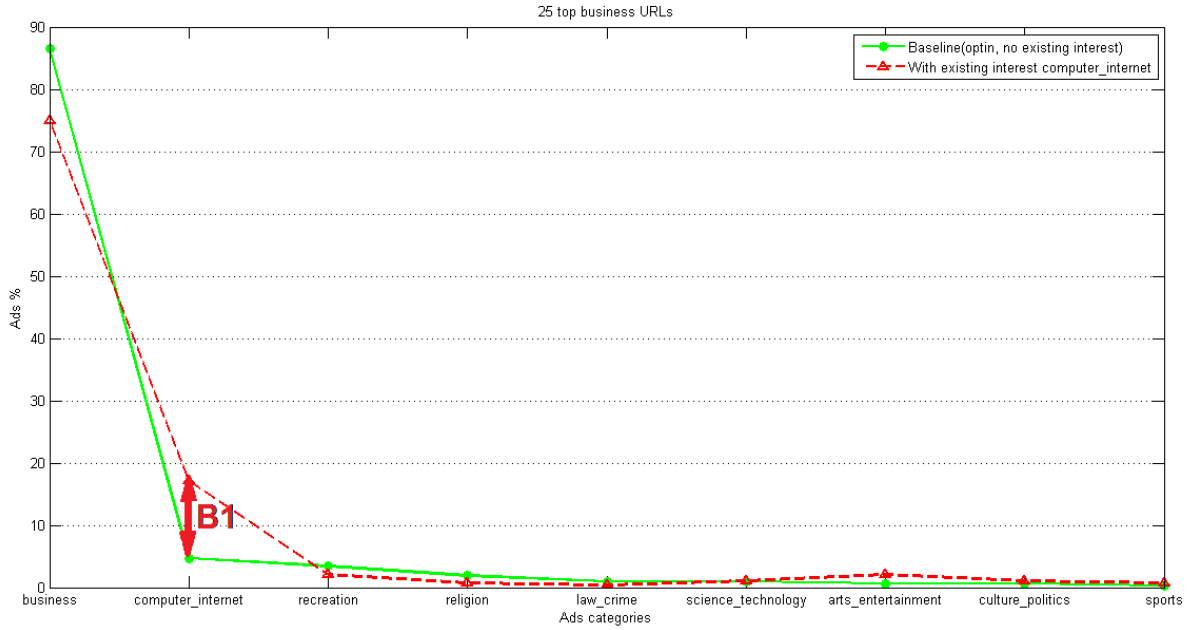


Figure A.2: Ads distribution when browsing URLs of category business with existing interest computer_internet, compared to the category baseline.

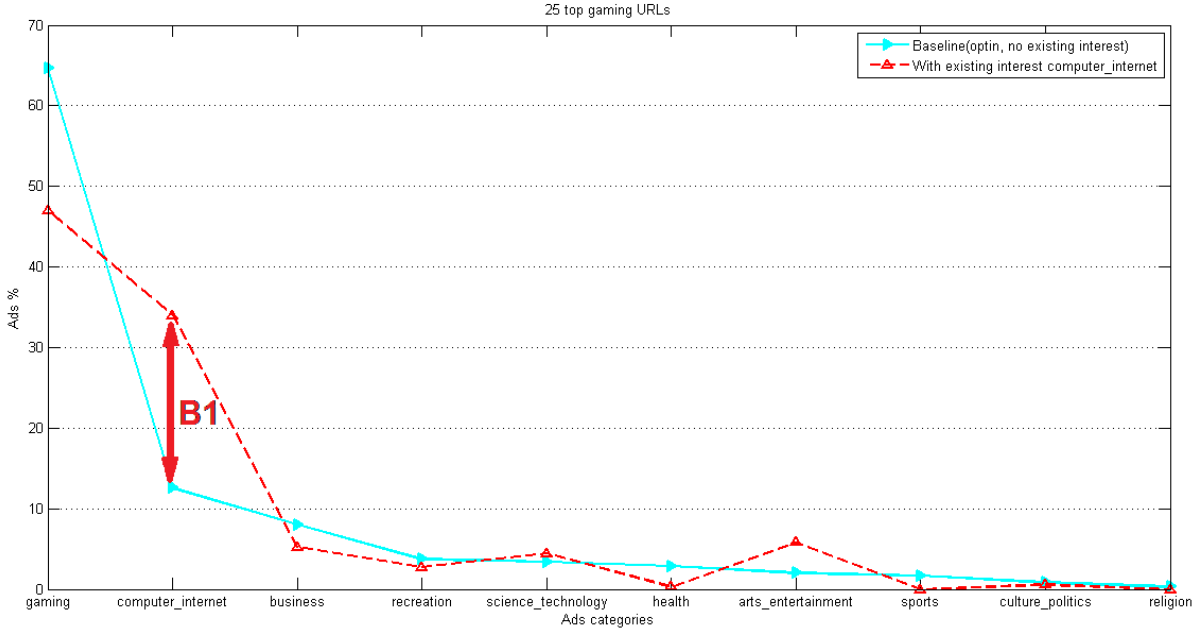


Figure A.3: Ads distribution when browsing URLs of category gaming with existing interest computer_internet, compared to the category baseline.

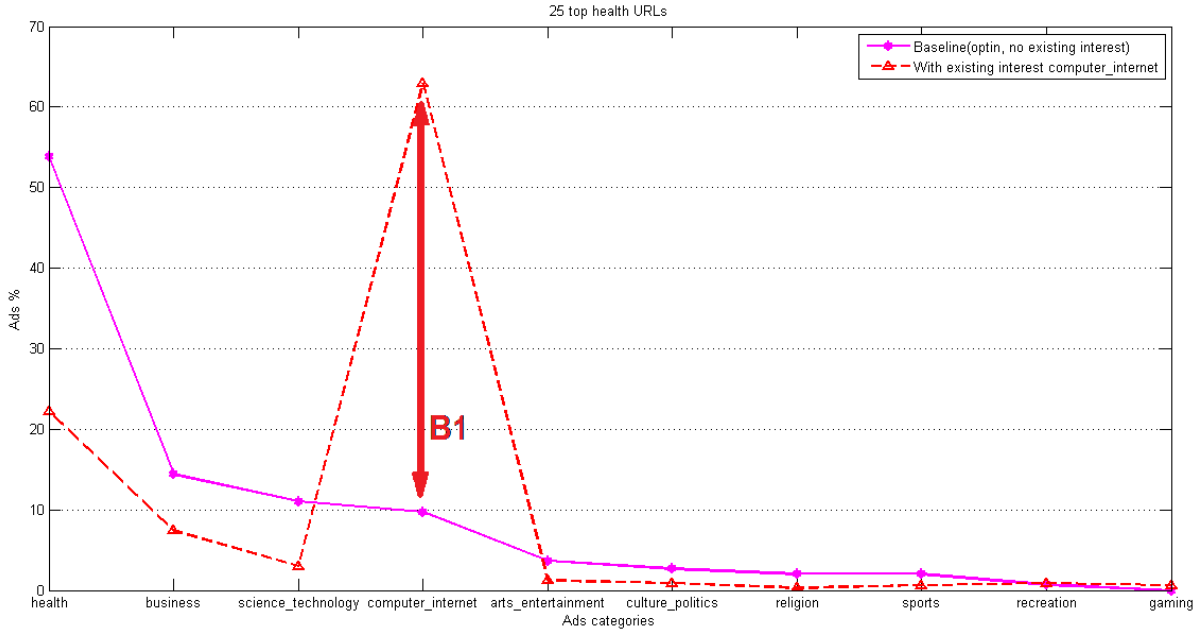


Figure A.4: Ads distribution when browsing URLs of category health with existing interest computer_internet, compared to the category baseline.

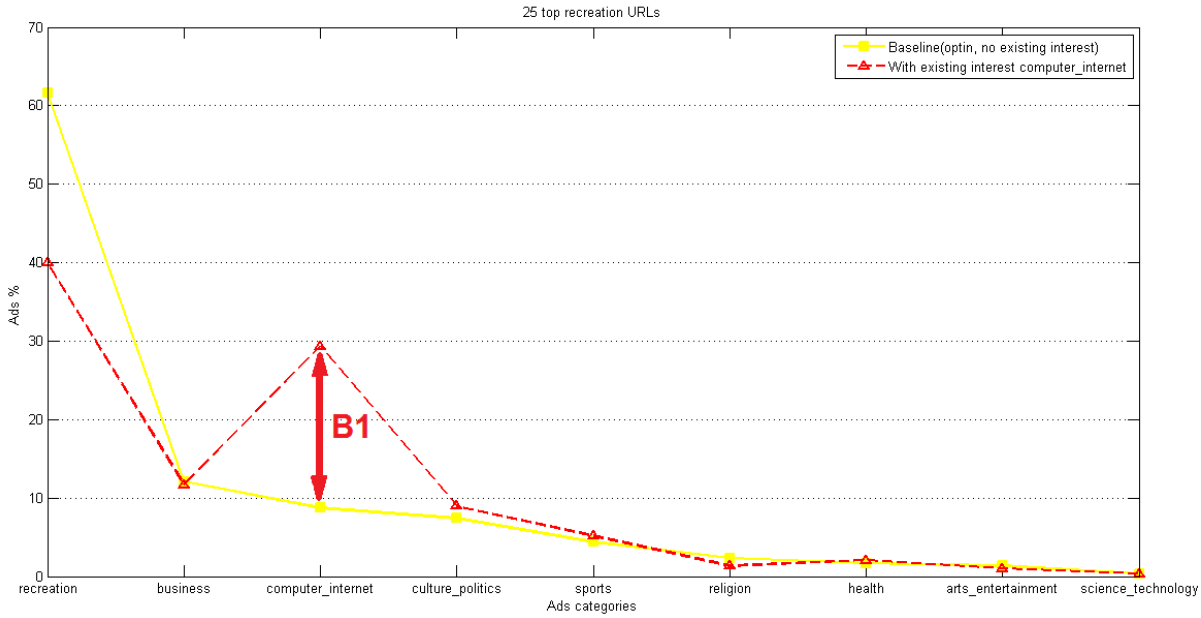


Figure A.5: Ads distribution when browsing URLs of category recreation with existing interest computer_internet, compared to the category baseline.

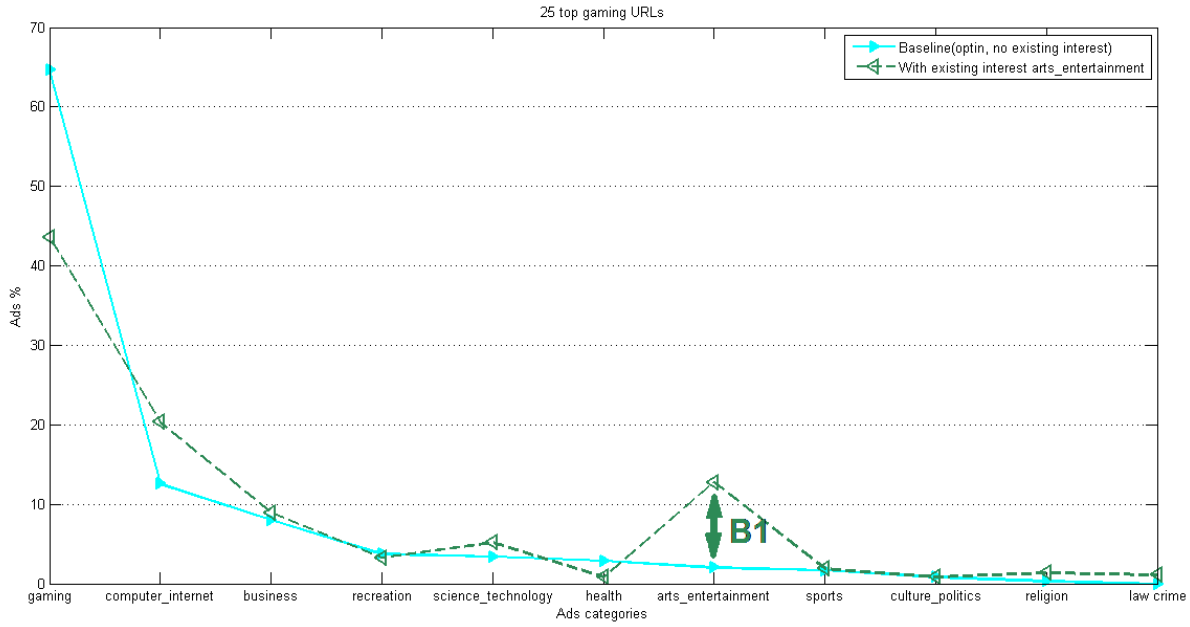


Figure A.6: Ads distribution when browsing URLs of category gaming with existing interest art_entertainment, compared to the category baseline.

Bibliography

[1] http://www.truste.com/ad-privacy/ (ref on 08.06.2012). Online privacy study.

[2] www.google.com/ads/preferences/. The Google Ads Preferences Manager.

[3] http://s3.amazonaws.com/alexa-static/top-1m.csv.zip (ref May 2012). Alexa globally ranked list of the top 1000000 websites in the world.

[4] http://www.alexa.com/topsites/category. Alexa categorized listing of top websites.

[5] http://www.alchemyapi.com/api/categ/. Alchemy API.

[6] 128.178.151.95. Our database EPFL server.

[7] Goldstein D Johnson EJ. ⟨ do defaults save lives? ⟩. *Science 302*, pages 1338–1339, 2003.