ARES – CONSOLIDER INGENIO 2010
CSD2007-00004

W3.T3: Development of privacy preserving transaction logs

G. Navarro-Arribas[1], J. Castellà-Roca[2]
[1] Institut d'Investigació en Intel·ligència Artificial,
Consejo Superior de Investigaciones Científicas,
Campus de la UAB, E-08193 Bellaterra, Catalonia (Spain)
e-mail guille@iiia.csic.es
[2] Universitat Rovira i Virgili,
Dept. d'Enginyeria Informàtica i Matemàtiques,
Av. Països Catalans 26, 43007 Tarragona, Catalonia (Spain)
e-mail jordi.castella@urv.cat

October 5, 2009

Contents

1 Introduction **1**

2 Client-side anonymization **3**
2.1 Pseudo identity 4
2.2 Group identity 4
2.2.1 Proxy approach 4
2.2.2 Obfuscation methods 4
2.2.3 Submit queries generated by other users 5
2.3 No identity 6
2.4 No personal information 7
2.5 Straightforward ways to provide anonymity 7

3 Server-side anonymization **8**
3.1 Anonymization of Access Logs 9
3.1.1 Web mining and access log format 9
3.1.2 Microaggregation of Access Logs 10
3.2 Anonymization of Search Logs 11
3.2.1 Background and Motivation 11
3.2.2 Microaggregation of Search Logs 12

4 Conclusions and contributions **14**
4.1 Conclusions 14
4.2 List of contributions 15
4.2.1 ISI JCR Journals 15
4.2.2 LNCS 15
4.2.3 Conferences and non ISI-JCR Journals 15

Abstract

When we do any transaction in Internet we leave a trail of our actions, which may lead to unparalleled invasions of our privacy. The queries that we sent to a web search engine (WSE) are an example. The WSEs create a user profile from our queries, which directly attempts against our privacy. The privacy protection in these scenarios can be addressed from two different points: the clients protect themselves without the WSE collaboration, or the WSE participates in the process of privacy protection. We have classified and described the main proposals in each point. Finally, we conclude with our published works in the two lines of research.

Chapter 1

Introduction

The privacy of users in electronic commerce, and more generally in any transaction made over the Internet, is a major concern for users themselves and social rights organizations. The users leave a trace of information resulting in mass collection of data, which companies tend to exploit using such tools as data warehouses.

One of the services massively used in the Internet are Web Search Engines (WSE). A WSE such as Google or Yahoo, collects and stores information of the searches made by users as search logs. This information is then used to create user profiles, which allow the WSE to provide a better personalized service or even personalized advertisement.

It is clear that the search logs generated by search engines are a great source of information for researchers or marketing companies, but at the same time their storage and publication exposes the privacy of the users from which the logs where generated. Even when this logs are poorly anonymized private information can be leaked.

There is at least one well known case of released search logs with poor anonymization, which have been shown to reveal enough information to reidentify some users. The release was done by AOL in an attempt to help the information retrieval research community, and ended up with not only important damage to AOL users privacy, but also a major damage to AOL itself with several class actions suits and complaints against the company [15, 31].

A part from WSE, other Web Services or generic customer and business interaction over the Internet present similar privacy concerns. From simply browsing the Web to full transaction log of online stores. Although most of the work presented here deals with anonymizing search logs, it can also be extended to protect other logs and online interactions.

The problem of anonymizing logs can be addressed from two different points:

1. Client-side anonymization: the anonymization is performed by the client (or user), in the moment of querying the WSE. The WSE does not collaborate in the anonymization process and it can even be considered as an adversary. The user attempts to hide its private information to the WSE itself, by using private information retrieval (PIR) protocols or other mechanism to query the WSE.

2. Server-side anonymization: in this case the anonymization process is performed a posteriori with the collaboration of the WSE. That is, as users make queries, the WSE stores a log of search queries/responses, which is latter anonymized by using for example, techniques from privacy preserving data mining (PPD) or statistical disclosure control (SDC). In this case, it is the WSE, which provides the anonymization of the search logs in order to either outsource marketing and profiling services, to make the information public to the research community, or as a requirement of data privacy protection laws.

In the following chapters we will outline our contributions in both cases.

Chapter 2

Client-side anonymization

Web search engines (WSE) provides to the users several result pages (web pages containing links to the resulting data). [24] states that 68% of the users of a web search engine click a search result within the first page of results. Also, 92% of the users click a result within the first three pages of search results. Therefore, in order to provide a better user experience, WSE should put in the first result pages the links that are more interesting for the users.

Nevertheless, it is not easy to know the users' interests. An example of that happens when looking for the word "Mercury". This term can refer to the planet Mercury or to an element in the periodic table. The concept *disambiguation* represents the process of identifying the correct sense when a certain word has different ones. The disambiguation process requires the knowledge of: (i) the interests of the user; or (ii) the query context. Both can be deduced from the user's profile which is built of her previous searches. For example, if a certain user has searched "solar system" before "Mercury", the web search engine will assume that "Mercury" refers to the planet Mercury and not to the element in the periodic table. According to that, the web search engine will put the results that correspond to the planet Mercury in the first pages.

In the literature, the process of improving the accuracy of the web search engines by profiling users receives the name of *personalized search (PS)* or *personalized web search (PWS)* [36, 25, 39, 42, 55].

How the users of web search engines are profiled is a topic which has been widely addressed in the literature. In [45], the authors use the browsing history. [38] proposes the use of click-though data. [43] introduces the use of web communities for this purpose. A client side application which stores users' interests is presented in [46]. However, the most successful approach for the web search engines is the use of the queries previously submitted by the users [43, 21]. This mechanism effectively profiles the users and it does not require their collaboration.

Even though the use of profiles improve the users' experience, they contain information that can be considered private and personal. If a certain user has searched for a certain place, it can be inferred that she lives there. If she looks for a certain disease, it can be deduced that she (or someone close to her) suffers that disease. These mechanisms put at risk the privacy of the users.

Note that profiles are needed in order to provide an efficient service to the

users. Thus, there is a trade-off between the privacy level achieved and the quality of the service. If the user desires a high degree of privacy, she will probably receive a deficient service. If the user desires an accurate service, her privacy will probably be jeopardized.

In [42], four levels of privacy protection are defined and analyzed: *pseudo identity*, *group identity*, *no identity* and *no personal information*.

2.1 Pseudo identity

In the *pseudo identity* level, the user identity is replaced by a pseudo-identity which contains less identifiable information. Nevertheless, this does not prevent the WSE from creating a profile associated to the pseudo-identity. This profile contains sensitive information that can be used to identify the real user who hides behind the pseudo-identity. For example, when AOL released its search engine log in 2006, they replaced IP addresses with pseudo-identities [3]. In that circumstance, this level of protection was proved to be insufficient.

2.2 Group identity

The second level of privacy corresponds to the *group identity* level. In this case, a group of users share a single identity. Therefore, the WSE is only able to build a group profile. It cannot profile single users. This mechanism improves the privacy level achieved by users. However, the use of a group profile instead of individual profiles reduces the effectiveness of the service.

There are three ways to implement this level of privacy: (i) using a proxy to construct the group; (ii) using an obfuscation mechanism like submitting random queries; and (iii) sending queries which have been generated by other users.

2.2.1 Proxy approach

One shortcoming of the proxy approach is that then the proxy can build the individual profiles like the WSE. Thus the proxy approach offers the lowest level of privacy.

2.2.2 Obfuscation methods

Regarding the use of obfuscation methods, submitting random queries misrepresent the profiles of the users. There are two works based on this solution that have been implemented in the Firefox web browser: TrackMeNot [52, 30] and GooPIR [13]. The authors in [42] state that this approach can be considered as a way to get k-anonymity [40]. Both TrackMeNot and GooPIR are fast because they do not create groups. However, they have the following drawbacks:

- TrackMeNot submits random queries to the web search engines when the users' activity is low. This is done to prevent the system from affecting the users' normal work. Nevertheless, sending fake queries increases the network traffic and overloads the WSEs. Therefore, this scheme protects

the privacy of the users, but it reduces the network and WSEs performance. In addition to that, this behavior introduces a serious privacy threat: for each user, the WSE is able to divide all her queries depending on whether or not they have been submitted during working hours (according to the time-zone of the user). Probably, all the queries submitted out of the working hours have been sent by TrackMeNot. The period of time between two different queries can also be used to the same purpose: it can be assumed that when the users are working, they do not submit only one query but several in a short period of time. This gap between queries can be used to deduce whether or not a certain query has been submitted by TrackMeNot.

- GooPIR submits fake words to the WSE together with the authentic one. This behavior obfuscates the user's profile because the WSE cannot know which words are fake and which are not. This proposal introduces a large overhead to neither the network nor the WSE. Nonetheless, GooPIR uses a Thesaurus in order to decide which words can be added to the search. According to that, GooPIR can only submit words. Full sentences are not addressed (note that sentences cannot be formed by random words).

2.2.3 Submit queries generated by other users

As explained previously, the last approach to implement the *group identity* level is based on users submitting queries generated by other users. There are two different proposals in the literature that follow this approach: [11, 12, 4].

The system proposed in [11, 12] uses memory sectors which are shared by a group of users. These users use the shared memory to store and read the queries and their answers. There is no connection between the users. Queries and answers are encrypted in order to provide confidentiality. This proposal does not require a trusted third party to create the groups or generate the cryptographic material. Instead of that,a simple wiki-like collaborative environment can be used to implement a shared memory sector. An onion routing protocol is used to preserve the privacy of the users that access this environment.

This scheme has the following drawbacks:

- It should be capable of managing a high volume of information. However, the memory-space requirements have not been studied by the authors.

- Users must scan their shared memory sectors at regular intervals. This requirement introduces a significant overhead to the network.

In [4], the Useless User Profile (UUP) protocol is proposed. The main idea of this scheme is that each user who wants to submit a query will not send her own query but a query of another user instead. Using this approach, the web search engine cannot generate a real profile of a certain individual. Regarding privacy concerns, the relevant point is that the users do not know which query belongs to each user. This is achieved using cryptographic tools. The system requires two components: a central node and a client application. The central node listens to client requests. After receiving n requests, it creates a new group and sends the IP addresses and port numbers to each group member. Then, the group members establish network connections between them and start to

communicate without the interaction of the central node. This scheme has been tested in real conditions and provides an overhead of 5.2 seconds with a group of three users and a key length of 1024 bits. The shortcomings of this proposal are the following:

- The groups of users must be created. This process introduces a significant delay.
- It requires a large number of users in order to provide an acceptable response time.

2.3 No identity

In the *no identity* level, the identity of the user is not available to the search engine. As a result, the WSE cannot profile the users. Almost all the proposals that fall in this level use an anonymous channel implementation. The *Tor Project* [47] is an example of that. There are several Firefox plugins [20, 48] that are based on Tor. This approach has two main drawbacks:

- If a user submits queries that contain personal information (*e.g.* name, social security number, city of residence...), her privacy vanishes.
- The process of submitting a query to the WSE and receiving the answer through an anonymous channel is very time-consuming. The authors in [39] used the anonymous network Tor with paths of length two (note that the default length is three) and submitting a query was, on average, 25 times slower than performing a direct search.

Private Information Retrieval (PIR) schemes also fall in the *no identity* level of privacy. PIR protocols protect the privacy of the users who retrieve information from servers. Thus, a PIR scheme enables the users to retrieve a certain value from a database while the server, which holds the database, gets no knowledge about the data requested by the users.

The first PIR protocol was designed by Chor, Goldreich, Kushilevitz and Sudan [5, 6] . This scheme is based on several severs holding the same database. The authors assume that those servers cannot communicate between them. This proposal cannot be used in scenarios with only one server (single-database PIR). Web search engines are an example of this type of scenarios.

Single-database PIR schemes can be found in [27] and [35]. Even though these proposals are suitable for WSEs, they have important shortcomings:

- The database is usually modeled like a vector. In that case, the user wants to retrieve the value stored in the i-th position of that vector. Both assumptions are not realistic because the database of the WSE is not a vector and the user does not know where the WSE stores the information.
- PIR schemes require that the WSE collaborates with the users. The WSE has no motivation to protect the privacy of the users. In addition to that, these schemes increase the computational and communication cost of the WSE.

2.4 No personal information

In the *no personal information* level, the identity of the user and the description of the data she desires are not available to the search engine. This level provides the highest privacy protection to the users. Nevertheless, the computational and communication costs required by these mechanisms make them unaffordable in practice [42].

2.5 Straightforward ways to provide anonymity

Note that, in addition to all the schemes explained above, there is a straightforward way to provide anonymity to the users who use WSEs: they can access to the Internet using a dynamic IP address and a controlled/clean web browser without cookies. However, this approach has the following drawbacks:

- The renewal policy of the dynamic IP address is not controlled by the user but the network operator. This operator can always give the same IP address to the same Media Access Control (MAC) address.
- Certain users require static IP addresses.
- A browser without cookies losses its usability in a high number of web applications. This situation may not be affordable for certain users.

Chapter 3

Server-side anonymization

The anonymization process in performed once the logs are generated by the WSE, or in general by any online service. That is, the logs are processed by an anonymization process so then they can be stored for future analysis, outsourced to marketing and profiling companies, or even made public.

We have addressed the anonymization of search logs and Web access logs. The last ones are those generate by any Web server, either hosting a simple Web page or more elaborated services such as online stores.

As we will see, we propose the application of microaggregation to anonymize both search logs and access logs. This approach ensures a high degree of privacy, providing k-anonymity, while preserving some of the data usefulness. Moreover it is the first time that statistical disclosure (SDC) techniques are being applied for such scenarios.

Microaggregation was originally [10] defined for numerical attributes, but later extended to other domains, e.g., to categorical data in [49], and in constrained domains in [50].

From the operational point of view, microaggregation is defined in terms of partition and aggregation:

- **Partition.** Records are partitioned into several clusters, each of them consisting of at least k records.
- **Aggregation.** For each of the clusters a representative (the centroid) is computed, and then original records are replaced by the representative of the cluster to which they belong to.

As the solution of this problem for multivariate records is NP-Hard [34] when we consider more than one variable at a time (multivariate microaggregation), heuristic methods have been developed.

MDAV (Maximum Distance to Average Vector) is one of such existing algorithms. The implementation of MDAV based on fuzzy clustering is given in [51], and MDAV for categorical data in in [14].

Note that when all variables are considered at once, microaggregation is a way to implement k-anonymity [40].

3.1 Anonymization of Access Logs

A very important use of Web access logs is to provide means for usage (and user) profiling of the site hosted by the given Web server. Usage profiling in such scenarios is normally done through *Web mining* [17, 7], by gathering information from the usage of the Web site (or service) to later on analyze it and come up with usage patterns. More precisely, these specific data mining approaches to determine the usage of a web site are normally denoted as *Web usage mining*, and complemented with what is known as *Web structure mining* and *Web content mining* [19, 26].

Web usage mining normally comprises two main techniques: statistical analysis, and more advanced data mining algorithms such as association rules, sequential patterns, and clustering [23]. While the first approach provides common and consolidated statistical estimations of the usage, the second approach attempts to identify usage patterns. In any case, both approaches rely on huge amounts of data gathered mainly from three sources: web servers, proxy servers, and web clients [19], although other sources such as specific application servers, or even application data can also be used [44]. The need to maintain such data in order to later on use them or even share them with other companies by outsourcing the usage profiling makes privacy issues very important.

The data used in Web usage mining contains sensitive information regarding the individual users, such as IP addresses. Moreover, Web usage mining and user profiling have always aroused privacy concerns among the users, governments, and civil rights organizations [41, 16, 18, 53]. Even so, there are not much proposals to effectively provide a degree of privacy in Web usage mining data.

3.1.1 Web mining and access log format

Web usage mining relies on both statistical analysis and data mining techniques, but a fundamental previous step is the data preprocessing [8], which does an initial analysis of the data gathered from the available sources.

One of the main concerns in the data preprocessing and Web usage mining is to be able to identify *sessions*. A session is the set of pages that a single users browses in one visit, normally with the associated time spent in each page.

The server logs are usually presented in a standard format known as Common Log Format [29] (CLF), or the Extended Log Format [8]. The CLF has the following information:

- *Client*: the IP address of the client.
- *rfc931*: remote logname of the user (not used or unreliable).
- *Authuser*: user name if HTTP authentication is used.
- *Date*: date and time of the client request.
- *Request*: the full request.
- *Status*: status code returned by the server.
- *Size*: size in bytes of the response.

which is normally complemented, for example in the default Combined Log Format [2] of the Apache HTTP server with:

- *User-agent*: browser and OS name and version of the client.
- *Referrer*: page preceding the one requested.

```
10.0.0.1 - - [11/Dec/2008:16:01:22 +0100] "GET /guille/index.html HTTP/1.1" 200
958 "http://hacks-galore.org/guille/others.html"
    "Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.5; en-US; rv:1.9.0.4)
Gecko/2008102920 Firefox/3.0.4"
10.0.0.1 - - [11/Dec/2008:16:01:23 +0100] "GET /guille/research.html HTTP/1.1" 200
1030 "http://hacks-galore.org/guille/index.html"
    "Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.5; en-US; rv:1.9.0.4)
Gecko/2008102920 Firefox/3.0.4"
```

Figure 3.1: Example of Web server log.

Fig. 3.1 shows an example of the format produced by the default Combined Log from the Apache HTTP server. It has two entries with the previously mentioned information (IP addresses have been changed).

3.1.2 Microaggregation of Access Logs

Our main concern when considering the microaggregation of the Web log data is to be able to produce useful data. In order to do that we must guarantee, that in the protected data:

1. Statistical information is maintained.
2. Data mining algorithms will give similar results.

It is well known that microaggregation gives good results maintaining general statistics, but the case of keeping the same results when applying data mining algorithms is a bit trickier.

On one hand most data mining algorithms applied to Web usage mining end up forming clusters or groups of related queries. It seems then that the groups formed by microaggregating the data in a previous step will not affect the final clustering. But to make it more anonymous we have considered the following requirement: *clusters formed in the microaggregation should be composed (as much as possible) of queries that do not correspond to the same* session. This requirement attempts on one hand to avoid having all the queries of the same session (and thus, user) in the same cluster, an that in the Web usage mining preprocessing step, we will find the same sessions as if the data were not protected.

To do that we microaggregate all the variables, except the IP. The IP address is considered and identifier and it is protected by substituting the IP itself by its hash (using a cryptographic one-way hash function). This ensures that sessions will most likely to be recovered while the IP address is unknown. We add the DNS name to the dataset, so one can obtain common information related to the location and administration domain of the request without the need to use the IP address.

To achieve the desired results we have defined a custom distance function and aggregator operator to microaggregate the Web logs. Such function and aggregator are defined in [32]. Moreover, the results obtained and evaluation of our proposed method are being submitted to an international journal.

3.2 Anonymization of Search Logs

In the first case, search logs generated by search engines are a great source of information for researchers or marketing companies, but at the same time their publication may expose the privacy of the users from which the logs where generated. Recall the AOL's case outlined in Section 1.

Achieving a desirable degree of privacy in search logs is not easy, and presents a trade-off between privacy and the usefulness of the data. There are several approaches [9] to anonymize such data but as far as we know, common techniques used in statistical disclosure control have not been applied to this specific problem.

3.2.1 Background and Motivation

The format and information contained in a search log may vary between different search engines. We have chosen a generic format, like the logs disclosed by AOL, and AllTheWeb, and which closely resemble other publicly available logs from Excite or AltaVista. The log has the form:

$$\{id, \textit{query-terms}, \textit{timestamp}, \textit{clickedURL}\} \tag{3.1}$$

Where *id* is the user identifier, *query-terms* is the query string (seen as a list of terms), *timestamp* is the time of the query, and *clickedURL* is the URL clicked by the user after the query. Other logs may include other information, for example the rank of the clicked URL (case of AOL), although this information may be obtained afterwards by re-running the query in the search engine (with some inaccuracies due to the time elapsed since the original query).

```
87b7c2, "data privacy", 2009−03−17 17:07:11, en.wikipedia.org
87b7c2, "data security", 2009−03−17 17:16:53, en.wikipedia.org
87b7c2, "privacy security journal", 2009−03−17 17:18:21, www.tdp.cat
5e7b6c, "privacy", 2009−03−17 17:20:50, en.wikipedia.org
5e7b6c, "transactions data privacy", 2009−03−17 17:21:47, www.tdp.cat
```

Figure 3.2: Example of search query log.

Normally the *clickedURL* is truncated to the domain name before publishing as a minor privacy measure. We can also assume that private information from the query terms such as social security numbers has been removed [54].

We also assume that the *id* of the user is a unique identifier for each user. This identifier can be obtained directly by engines where the user needs to log in, or indirectly by, for example, as a combination of the URL, user agent, and cookies of the user. This *id* is normally anonymized by a simple hash or a similar aproach. It has been shown that, even using such anonymization, users

can be identified [3]. Moreover, hashing techniques, applied to the query terms, are vulnerable to frequency analysis [28].

Other anonymization techniques have been developed for search logs, such as removing infrequent queries [1], or more sophisticated techniques to remove selected queries which do not preserve an acceptable degree of privacy [37].

3.2.2 Microaggregation of Search Logs

One may attempt to microaggregate a search log directly. That is, where each record corresponds to each line in the log. In this case, the *id* can be considered an identifier and it won't be microaggregated (it is removed or encrypted) using whatever distance function and aggregate operator is required. The problem is that the final protected data will suffer privacy problems due to the fact that the *id* can be repeated among several records. If we remove the *id*, the data looses much of its utility, while if we keep it (even hashed or encrypted) depending on how the clusters are formed there may be a considerable risk of re-identification.

In order to avoid such problem we opt to include the *id* in the microaggregation. The *id* will be microaggregated together with the other variables. Furthermore we use a tree-based structure to represent the logs belonging to the same *id* and then microaggregate the trees as if they were a record of microdata in common microaggregation.

All log queries with the same *id* are represented as an ordered tree of three levels: the root of the tree is the *id*, the level 1 nodes are the queries, represented by the timestamp and the *clickedURL*, and the level 2 are the search terms grouped by query. For example, Figure 3.3 shows the tree representation of the query logs from Tab. 3.1, which represent the log of Figure 3.2.

id	*query terms*	*timestamp*	*clickedURL*
id_0	$\{\mu_0, \mu_1\}$	t_0	U_0
id_0	$\{\mu_0, \mu_2\}$	t_1	U_1
id_0	$\{\mu_1, \mu_2, \mu_3\}$	t_2	U_1
id_1	$\{\mu_1\}$	t_3	U_2
id_1	$\{\mu_4, \mu_0, \mu_1\}$	t_4	U_3

Table 3.1: Example of queries.

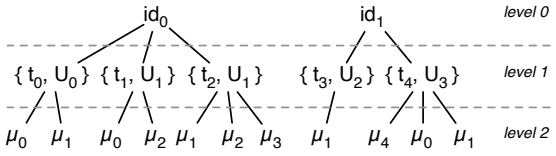

Figure 3.3: Example of query trees.

In these examples, *id* denotes a protected (hashed or encrypted) user identifier, μ_i is a query term, t_i a timestamp, and U_i an URL.

We call this kind of tree a *query tree*. The tree is *ordered* from left to right at each level. In order to use microaggregation for *query trees* we only need to define a *distance* function, and an *aggregator* operator.

Finally, in order to microaggregate *query trees* we define an specific distance funtion and aggregation operator, which can be found in [33].

Chapter 4

Conclusions and contributions

4.1 Conclusions

We have presented the privacy threat to which users are exposed when they sue a Web Search Engine (WSE). The WSEs create the users profiles using the searches made by them. Thus, the WSE can provide a better personalised service or even personalized advertisement.

Next, we have described two different approaches to address the problem of anonymizing logs: client-side anonymization and server-side anonymization.

In the client-side anonymisation, the anonymisation is performed by the clients (or users), when they send their queries to the WSE. In this case, the WSEs have no interest in collaborating with the anonymisation process. The proposals can be classified in four levels [42]: *pseudo identity*, *group identity*, *no identity* and *no personal information*. The *pseudo identity* does not offer enough privacy protection, and the *no identity* and *no personal information* proposals are not usable in practice due their high response time. Thus, our work has focused on the *group identity* level, publishing four proposals [13, 11, 12, 4], three in ISI JCR journals [13, 12, 4] and one in an international conference [11]. GooPIR [13] is a new obfuscation method, and [11, 12, 4] proposals protect the users' privacy submitting queries generated by other users. We are working in new proposals in the same directions: obfuscation and submitting queries generated by others users.

In the server-side anonymization, we have outlined our main work towards the anonymisation of logs generated by Web server and WSE by making use of microaggregation. We are not aware of any SDC technique used in such field, and our results are quite promising. The results of the work presented haven been published in two international conferences [32, 33], and are currently being submited to international journals.

As a result of this work, we are currently applying similar techniques to anonymize full transaction logs from e-comerce sites. Moreover we are working in the anonymization of the logs generated by the popular trafic analysis software Urchin from Google [22]. This format known as ELF2 (*E-commerce Log Format 2*) contains information about each order placed by the user separately,

containing all the items of the purchase. It is one of the most used formats in current e-comerce sites.

4.2 List of contributions

A list of papers published or submitted by members of the ARES group follows.

4.2.1 ISI JCR Journals

1. J. Castellà-Roca, A. Viejo, J. Herrera-Joancomartí, "Preserving user's privacy in web search engines", *Computer Communications*, vol. 32, no. 13–14, pp. 1541–1551, 2009.

2. J. Domingo-Ferrer, M. Bras-Amorós, Q. Wu, J. Manjón, "User-Private Information Retrieval Based on a Peer-to-Peer Community", *Data and Knowledge Engineering*, to appear.

3. J. Domingo-Ferrer, A. Solanas, J. Castellà-Roca, "$h(k)$-Private Information Retrieval from Privacy-Uncooperative Queryable Databases", *Journal of Online Information Review*, vol. 33, no. 44, pp. 1468–4527, 2009.

4. G. Navarro-Arribas, V. Torra. "Microaggregation of log files: privacy for e-commerce". *Submitted*.

5. A. Viejo, J. Castellà-Roca, "Using Social Networks to Distort Users' Profiles Generated by Web Search Engines", *Submitted*.

4.2.2 LNCS

6. J. Domingo-Ferrer and M. Bras-Amorós. "Peer-to-peer private information retrieval". In *Privacy in Statistical Databases-PSD2008*, LNCS5262. Springer- Verlag, pp. 315-323, 2008.

4.2.3 Conferences and non ISI-JCR Journals

7. G. Navarro-Arribas, and V. Torra. Towards microaggregation of log files for Web usage mining in B2C e-commerce. In *28th North American Fuzzy Information Processing Society Annual Conference. Special session: Intelligent E-Services and Multi-Agent Systems for Web-Based B2C E-Commerce*. 2009.

8. G. Navarro-Arribas, and V. Torra. Tree-based Microaggregation for the Anonymization of Search Logs. In *WI/IAT'09 Workshop on Soft approaches to information access on the Web*. 2009.

9. I. Cano, G. Navarro-Arribas, V. Torra. A new framework to automate constrained microaggregation. In *ACM First International Workshop on Privacy and Anonymity for Very Large Datasets. To appear*. 2009.

10. G. Navarro-Arribas and J. Garcia-Alfaro. A policy based approach for the management of Web browser resources to prevent anonymity attacks in Tor. *24th IFIP TC-11 International Information Security Conference*

(SEC2009). Vol. 297 of IFIP Advances in Information and Communication Technology, 2009.

11. V. Torra. "Constrained Microaggregation: Adding Constraints for Data Editing". *Transactions on Data Privacy*, vol. 1, no. 2, pp. 86–104, 2008.

Bibliography

[1] E. Adar. User 4xxxxx9: Anonymizing query logs. In *Query Logs workshop*, 2007.

[2] The Apache Software Foundation. Log files. Apache HTTP Server Version 2.2 Documentation, 2009. URL `http://httpd.apache.org/docs/2.2/logs.html`.

[3] M. Barbaro and T. Zeller. A face is exposed for AOL searcher no. 4417749. *The New York Times*, Aug. 2006.

[4] J. Castellà-Roca, A. Viejo, J. Herrera-Joancomartí, "Preserving user's privacy in web search engines", *Computer Communications*, vol. 32, no. 13–14, pp. 1541–1551, 2009.

[5] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, "Private information retrieval", *IEEE Symposium on Foundations of Computer Science – FOCS*, pages 41–50, 1995.

[6] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, "Private information retrieval", *Journal of the ACM*, vol. 45, pp. 965–981, 1998.

[7] R Cooley, B Mobasher, and J Srivastava. Web mining: information and pattern discovery on the world wide web. *Tools with Artificial Intelligence*, January 1997.

[8] R Cooley, B Mobasher, and J Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, January 1999.

[9] A. Cooper. A survey of query log privacy-enhancing techniques from a policy perspective. *ACM Transactions on the Web*, 2(4), 2008.

[10] D. Defays and P. Nanopoulos. Panels of enterprises and confidentiality: the small aggregates method. In *Proc. of 92 Symposium on Design and Analysis of Longitudinal Surveys, Statistics Canada*, 1993.

[11] J. Domingo-Ferrer and M. Bras-Amorós. "Peer-to-peer private information retrieval". In *Privacy in Statistical Databases-PSD2008*, LNCS5262. Springer- Verlag, pp. 315-323, 2008.

[12] J. Domingo-Ferrer, M. Bras-Amorós, Q. Wu, J. Manjón, "User-Private Information Retrieval Based on a Peer-to-Peer Community", *Data and Knowledge Engineering*, to appear.

[13] J. Domingo-Ferrer, A. Solanas, J. Castellà-Roca, "$h(k)$-Private Information Retrieval from Privacy-Uncooperative Queryable Databases", *Journal of Online Information Review*, vol. 33, no. 44, pp. 1468–4527, 2009.

[14] J. Domingo-Ferrer and V. Torra. Ordinal, continuous and heterogeneous k-anonymity through microaggregation. *Data Mining and Knowledge Discovery*, 2005.

[15] EFF. AOL's massive data leak. Electronic Frontier Foundation, `http://w2.eff.org/Privacy/AOL/`, (2009).

[16] M Eirinaki and M Vazirgiannis. Web mining for web personalization. *ACM Transactions on Internet Technology*, January 2003.

[17] Oren Etzioni. The world-wide web: quagmire or gold mine? *Communications of the ACM*, 39(11), November 1996.

[18] EU. Directive 94/46/ec of the european parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal of the European Community (281), 1995.

[19] F Facca and P Lanzi. Mining interesting knowledge from weblogs: a survey. *Data & Knowledge Engineering*, January 2005.

[20] Foxtor, 2009. `http://cups.cs.cmu.edu/foxtor`

[21] Google personalized search, 2009. `http://www.google.com/psearch`

[22] Google. Urchin software from google. `http://www.google.com/urchin/index.html`, 2009.

[23] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, January 2006.

[24] iProspect.com, Inc,. "iProspect Blended Search Results Study", 2008. `http:// www.iprospect.com`

[25] G. Jeh, J. Widom, "Scaling personalized web search", *Proc. of the 12th International World Wide Web Conference*, 2003.

[26] R Kosala and H Blockeel. Web mining research: a survey. *ACM SIGKDD Explorations Newsletter*, January 2000.

[27] E. Kushilevitz, R. Ostrovsky, "Replication is not needed: single database, computationally-private information retrieval", *Proc. of the 38th Annual IEEE Symposium on Foundations of Computer Science*, pp. 364–373, 1997.

[28] R. Kumar, J. Novak, B. Pang, and A. Tomkins. On anonymizing query logs via token-based hashing. In *16 International World Wide Web Conference.*, 2007.

[29] A. Luotonen. The common logfile format, 1995. URL `http://www.w3.org/pub/WWW/Daemon/User/Config/Logging.html`.

[30] G. T. Marx, "A Tack in the Shoe: Neutralizing and Resisting the New Surveillance" *Journal of Social Issues*, vol. 59, no. 2, pp. 369–390, 2003.

[31] E. Mills. AOL sued over web search data release. CNET News, `http://news.cnet.com/8301-10784_3-6119218-7.html`, Sept. 2006.

[32] G. Navarro-Arribas, and V. Torra. Towards microaggregation of log files for Web usage mining in B2C e-commerce. In *28th North American Fuzzy Information Processing Society Annual Conference. Special session: Intelligent E-Services and Multi-Agent Systems for Web-Based B2C E-Commerce*. 2009.

[33] G. Navarro-Arribas, and V. Torra. Tree-based Microaggregation for the Anonymization of Search Logs. In *WI/IAT'09 Workshop on Soft approaches to information access on the Web*. 2009.

[34] A. Oganian and J. Domingo-Ferrer. On the complexity of optimal microaggregation for statistical disclosure control. *Statistical Journal of the United Nations Economic Commision for Europe*, 18(4), 2001.

[35] R. Ostrovsky, W. E. Skeith-III, "A survey of single-database pir: techniques and applications", *Lecture Notes in Computer Science*, vol. 4450, pp. 393–411, 2007.

[36] J. Pitkow, H. Schuetze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, T. Breuel, "Personalized search", *Communications of the ACM*, vol. 45, no. 9, pp. 50–55, 2002.

[37] B. Poblete, M. Spiliopoulou, and R. Baeza-Yates. Website privacy preservation for query log publishing. In *First International Workshop on Privacy, Security, and Trust in KDD (PinKDD 2007)*, 2008.

[38] F. Qiu, J. Cho, "Automatic identification of user interest for personalized search", *Proc. of the 12th International World Wide Web Conference*, 2006.

[39] F. Saint-Jean, A. Johnson, D. Boneh, J. Feigenbaum, "Private Web Search", *Proc. of the ACM workshop on Privacy in electronic society – WPES'07*, pp. 84–90, 2007.

[40] P. Samarati. Protecting Respondents' Identities in Microdata Release. *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 6. November/December, 2001.

[41] J Schafer, J Konstan, and J Riedl. E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, January 2001.

[42] X. Shen, B. Tan, C.X. Zhai, "Privacy Protection in Personalized Search", *ACM SIGIR Forum*, vol. 41, no. 1, pp. 4–17, 2007.

[43] M. Speretta, S. Gauch, "Personalizing search based on user search history", *Proc. of International Conference of Knowledge Management –CIKM'04*, 2004.

[44] J Srivastava, P Desikan, and V Kumar. Web mining-concepts, applications and research directions. *Studies in Fuzziness and Soft Computing*, January 2005.

[45] K. Sugiyama, K. Hatano, M. Yoshikawa, "Adaptive Web Search based on user profile constructed without any effort from users", *Proc. of the 13th International World Wide Web Conference*, 2004.

[46] J. Teevan, S. T. Dumais, E. Horvitz, "Personalizing search via automated analysis of interests and activities", *Proc. of 28th Annual International ACM Conference on Research and Development in Information Retrieval –SIGIR'05*, 2005.

[47] The Tor Project, 2009. `http://www.torproject.org`

[48] Torbutton, 2009. `http://freehaven.net/~squires/torbutton`

[49] V. Torra. Microaggregation for categorical variables: A median based approach. In *Proc. Privacy in Statistical Databases (PSD 2004)*, 2004.

[50] V. Torra. Constrained microaggregation: Adding constraints for data editing. *Transactions on Data Privacy*, 1(2), 2008.

[51] V. Torra and S. Miyamoto. Evaluating fuzzy clustering algorithms for microdata protection. In *Privacy in Statistical Databases (PSD2004)*, 2004.

[52] TrackMeNot, 2009. `http://mrl.nyu.edu/ dhowe/trackmenot`

[53] W3C. Platform for privacy preferences (p3p) project, 2008. URL `http://www.w3.org/P3P/`.

[54] L. Xiong and E. Agichtein. Towards privacy-preserving query log publishing. In *16 International World Wide Web Conference*, 2007.

[55] Y. Xu, B. Zhang, Z. Chen, K. Wang, "Privacy-Enhancing Personalized Web Search", *Proc. of the 16th international conference on World Wide Web*, pp. 591–600, 2007.