Providing Useful and Private Web Search by Means of Social Network Profiling

Alexandre Viejo, David Sánchez
Universitat Rovira i Virgili, UNESCO Chair in Data Privacy
Departament d'Enginyeria Informàtica i Matemàtiques
Av. Països Catalans 26, E-43007 Tarragona, Spain
Corresponding author email: alexandre.viejo@urv.cat

***Abstract*—Web search engines (WSEs) build user profiles and use them to offer an enhanced web search experience. Nevertheless, these elements might contain sensitive data that may represent a privacy threat for the users. There are some works in the literature that address this situation while preserving the profile usefulness. These schemes submit synthetic queries that are fake but related to the real general interests of the user. Specifically, they rely on past user queries to obtain the legitimate interests of each user. We argue that this is not always the best strategy and, in this paper, we study the use of social networks to gather this information and provide a better personalized service while offering an equivalent privacy level.**

I. Introduction

A *query log* is a collection of data that contains the list of queries submitted by each user of a certain web search engine (WSE). In general, these data bases are used to improve the accuracy and efficiency of web information retrieval. In this paper, we stress their key role in profiling users (i.e., building *user profiles* that contain their interests).

User profiles are essential to provide *personalized web search*. This technique can be used to provide an enhanced web search experience to the users by means of services such as *query disambiguation* or *query refinement* among others. On the other hand, WSEs may exploit their generated profiles to get important economical revenues. More specifically, they can be used to sell personalized advertising or they can also be directly sold to law enforcement agencies [1].

The possibility that those generated user profiles might be linked with real users implies a significant privacy cost that is generally neglected. However, some situations reported in the literature show that this probability is quite realistic. More specifically, the authors in [2] were able to identify a certain user using a query log of 20 million queries made by 658000 users which was publicly disclosed by AOL.

As a conclusion, new privacy-preserving tools should be designed to restrict the specific information that appears in these user profiles. Nevertheless, profiles are needed to provide an efficient personalized service to users. Therefore, any solution should address both privacy and usability requirements.

A. Previous work

In general, all the schemes that try to provide private web search follow two main approaches: (i) Conceal the real identity of the user in front of the WSE; and (ii) Distort the user profile by submitting *fake* queries together with *legitimate* ones to the WSE. Schemes based on the first approach do not allow WSEs to build any profile. Therefore, they are very good from the privacy point of view but they are not valid for providing personalized services. On the other hand, the second approach permits WSEs to build distorted profiles that contain a mix of *fake* and *legitimate* interests. In this way, they provide a trade-off between privacy and profile usefulness and, hence, the rest of this document focuses on them. Note that this kind of protocols can, in turn, be divided into two subcategories: *multi-party* and *single-party*.

Multi-party protocols require the collaboration of external entities and, for this reason, they suffer from *slow response time* (i.e., schemes such as [3] or [4] report response times between 3 and 6 seconds). Regarding the *single-party protocols*, these systems work directly in the computer of the user to be protected and they do not require a direct interaction with any external party. As a result, these schemes are ideally suited to provide the best response time and they completely control the content of the queries which are submitted to the WSE. According to these benefits, this paper focuses on single-party schemes.

TrackMeNot [5] and GooPIR [6] are well-known single-party schemes that generate and submit random *fake* queries. As a result, both methods generate profiles that contain a mix of authentic and random interests and, hence, WSEs will be unable to provide a satisfactory personalized service. In contrast, works such as [7] and [8] acknowledge the importance of providing privacy but also quality of service. In order to achieve that, these proposals consider the semantic distance between *legitimate* and *fake* queries. In this way, each *fake* query is synthetically generated according to the profile of their *legitimate* queries. Their main target is to force the WSE to build a specific profile with a specific level of detail that preserves the privacy of the user while being still useful.

The main hypothesis of [7] and [8] is that past queries effectively reflect the real interests of the user and, hence, they can be used to build *local profiles*. However, we believe that this assumption does not always hold due to the following points: points: (i) Circumstantially, users may submit several queries related to a certain topic which is quite far from their real interests; (ii) Users can submit quite inaccurate queries to the WSE expecting better suggestions. These tryouts add an

erroneous bias to the generated profiles; (iii) If different users share the same web search system to submit their queries to the WSE, the local profile will contain the aggregated interests of all of them; and (iv) There is no pattern that defines the structure of a query and, hence, it can be quite challenging to extract accurate interests from this data [8].

According to these points, we argue that the quality of the personalized service offered by schemes that rely on past queries to perform the profiling process may be unsatisfactory. Therefore, we stress that other sources of data more suited to build local profiles should be investigated.

B. Contribution and plan of this paper

In this paper, we propose a new scheme for private web search that distorts user profiles by submitting *fake* queries to the WSE. The novelty comes from the fact that it locally profiles users according to their social network accounts (e.g., Twitter) instead of from their supposed past searches. We argue that this behaviour solves (or at least alleviates) the four problems related to [7] and [8] which have been stressed in the previous section.

The main target of our proposal is to force the WSE to build a user profile that contains the *macro-interests* of the user (i.e., general interests) but hiding her *micro-interests* (i.e., her detailed interests into each general category). Note that *macro-interests* can be useful to provide personalized web search and they do not disclose enough information to represent a major privacy concern. On the other hand, *micro-interests* offer more details and can be more harmful from the privacy point of view.

The rest of this paper is organized as follows. Section II explains the new scheme. Section III provides an evaluation of the system. Finally, Section IV gives some concluding remarks.

II. Our proposal

The proposed method follows two main steps: *building the local profile* and *generating a fake query*. In the first one, our method profiles the user according to her publications in a social network. More concretely, we focus on Twitter (a very well-known social network and microblogging site) and we apply the profiling method for this platform which was presented in [9]. This scheme profiles users using only their published tweets (i.e., text-based posts up to 140 characters). Regarding the second step, it assesses which are the dominant and dominated categories in her *local profile* and which should be the contents of the *fake* queries to be submitted to the WSE. Those fake queries must fulfill two requirements: (i) they must be plausible in order to effectively hide the *legitimate* ones; and (ii) they must force the WSE to build the desired *user profile*.

We next describe both steps in detail and also explain the life cycle of the proposed system.

A. Building the local profile

As explained above, in order to build the local profile, we rely on the profiling method presented in [9]. This scheme explicitly focuses on Twitter, but it could also be integrated with other social platforms that work with textual publications. We next give a brief description about this profiling process.

Let us consider that a *local profile* (LP) is defined as a set of well-defined categories $C = \{c_1, \ldots, c_k\}$ (*e.g.,* science, health, society, sports, etc), for which their relative weight (v_i) should be computed according to the number of evidences found, for each category, in user tweets. As a result, the local profile of a certain user (LP) can be characterized according to the set of weighted profile categories obtained from analyzing a set of her tweets: $LP = \{< c_1, v_1 >, \ldots, < c_k, v_k >\}$.

In order to calculate the corresponding weight v_i for each category c_i, three main steps are followed:

1) *Extract noun-phrases (NPs) from user tweets*. NPs are pieces of text with semantic content that have been used in the literature to characterize user profiles [10]. To extract them, we rely on several commonly-used natural language processing tools (e.g., OpenNLP Maxent Package): *sentence detection*, *tokenization*, *part-of-speech tagging* and *syntactic parsing* (*i.e.,* chunking). As a result, from each published tweet, the set $\{< NP_1, w_1 >, \ldots, < NP_p, w_p >\}$ is obtained, being w_i the number of appearances of the corresponding NP_i.
2) *Classify extracted NPs according to the defined categories*. To enable this classification from a semantic point of view (*i.e.,* to link each NP with its conceptual abstraction), the proposed system relies on a predefined knowledge base. More concretely, the Open Directory Project (ODP) is used. Therefore, to semantically classify NPs, the system queries each extracted NP_i to ODP. If found, ODP returns the most likely hierarchy H_i of categories ($H_i = h_{i,1} \rightarrow \ldots \rightarrow h_{i,l}$) to which NP_i belongs. For example, if the system queries the NP *"iPhone"*, ODP returns: *iPhone* $\rightarrow$ *Smartphones* $\rightarrow$ *Handhelds* $\rightarrow$ *Systems* $\rightarrow$ *Computers*.
3) *Update LP according to the categories linked to each NP*. For each NP_i, ODP has retrieved a hierarchy H_i of categories. Therefore, the system checks if any of the considered categories c_i is included in H_i. If this is the case, the contribution of NP_i is added to c_i by adding its number of occurrences (i.e., w_i) to the corresponding category weight (*i.e.,* v_i). As a result, each weight v_i states the sum of contributions according to the number of term occurrences/co-occurrences related to each particular profile category c_i.

B. Generating a fake query

The purpose of our system is to generate *fake* queries that, together with the *legitimate* ones, force the WSE to build a user profile similar to LP (similar categories and corresponding weights) due to the fact that, as explained previously, we assume that LP is a realistic representation of the general interests of the user. However, *fake* queries must

contain plausible contents capable of hiding the the *detailed* interests of the user (*legitimate* and *fake* interests must be indistinguishable). In this way, we guarantee that the WSE will be able to provide a still useful personalized service (we keep the macro-interests), while addressing the privacy dimension (the micro-interests are hidden).

In this step, we need a local representation of the user profile that, we assume, is being build by the WSE. Let us name it as *public profile* (PP). The proposed system builds the PP using the queries submitted to the WSE in a similar way that the WSE is assumed to build its own profiles using the received queries. Therefore, we believe that our local PP is a reasonable approximation to the user profile built by the WSE.

PP can be defined similarly to LP. Therefore, we consider that $PP = \{< c_1, v_1 >, \ldots, < c_k, v_k >\}$. Moreover, at the beginning of its execution, our system has no idea about the queries that have been previously submitted, hence, PP is initialized as $PP = \{< c_1, 0 >, \ldots, < c_k, 0 >\}$.

Then, to calculate the corresponding weight v_i for each category c_i in PP, the proposed system applies the same profiler which has been already used to build LP. Nevertheless, in this case, the input are web search queries instead of tweets. Therefore, each query submitted by the user is analyzed and according to the number of term occurrences/co-occurrences related to each particular profile category, the different weights are updated. Note that, both *legitimate* and *fake* queries (these are the queries generated in this step) are analyzed to build the PP.

Next, let us focus on the steps required to generate each *fake* query:

1) *Normalize weights for LP and PP*. Both profiles LP and PP are defined in a similar way (i.e., they share the same categories) but the order of magnitude for the different weights are expected to be quite different. Therefore, in order to compare them properly, all the weights v_i have to be translated into the same scale. In order to achieve that, we assign a percentage λ_i to each category c_i according to its specific number of term occurrences v_i with regard to the total number of term occurrences got by all the categories (i.e., $\sum_{j=1}^{k} v_j$). More formally, let us consider that LP' and PP' are the normalized versions of LP and PP respectively. According to that, LP' and PP' are defined as $\{< c_1, \lambda_1 > , \ldots, < c_k, \lambda_k >\}$ where $\lambda_i = (v_i / \sum_{j=1}^{k} v_j) \times 100$.
2) *Select the category of the new fake query*. LP' is assumed to reflect the optimal weight for each category while PP' is assumed to reflect the current disclosed weight for each one. Therefore, the target of the whole system is to approximate PP' to LP' by submitting new *fake* queries. The topic for each new *fake* query is decided by detecting which category of PP' is currently further away from its "optimal weight" (this is reflected by LP'). In this way, for each category c_i, the system computes the difference between its optimal weight and its current weight. Let us denote this as: $\Delta_{c_i} = (\lambda_i; LP') - (\lambda_i; PP')$. Finally, the category that has got the largest positive Δ is selected.
3) *Generate the new fake query*. At this point, the proposed system randomly retrieves a term from the ODP hierarchy under the category that was selected in the latter step. This term is the new *fake* query to be submitted to the WSE. Due to the fact that this term is related to the required category, it is assumed to approximate PP to LP. Moreover, its randomness is assumed to hide the *micro-interests* of the user. Note also that this step is done following the same process introduced in [8]. The authors of this work evaluated the indistinguishability of the generated *fake* queries and the privacy level achieved by the users. In both cases, the results achieved were positive.
4) *Update PP according to the category linked to the fake query*. Assuming that the category c_i was selected in the former steps, the system increases its related weight v_i in PP by one term.

C. System's life cycle

This section describes how the whole system behaves and how the two main operations (i.e., *building the local profile* and *generating a fake query*) are used together:

1) Initially, the *building the local profile* step is used to profile the user according to her Twitter account. As a result, her LP is generated. Note that, at this point, her PP is still empty.
2) When the user submits a *legitimate* query to the WSE, first, her PP is updated with the contents of her query. Then, the system uses the *generating a fake query* step to individually build t *fake* queries (their contents are also used to update PP).
3) The t *fake* queries are submitted to the WSE.
4) Every time the user submits a new *legitimate* query, the system goes back to step-2.

III. Evaluation

In this section we evaluate the performance of the proposed system in approximating PP to LP. Those profiles contain eight different categories: *Arts*, *Health*, *Shopping*, *Science*, *Computers*, *Sports*, *Society* and *Business*.

As test data, we considered a set ψ of user search queries that were used as *legitimate queries* in our evaluations. More specifically, set ψ contained 100 queries randomly selected from the AOL query logs. Each query in ψ was profiled using the same classifier introduced in Section II-A. Using ψ as input, the corresponding normalized weights related to each category are shown in Table I. This table represents the PP' that the WSE would build if a certain user submits the whole set ψ (and *only* this set).

Table I shows that the most representative category in ψ is *Arts*, while the least representative one is *Science*. In order to evaluate the performance of our scheme in both the best and the worst scenarios, we took two well-differentiated Twitter users whose normalized local profiles LP' presented a very

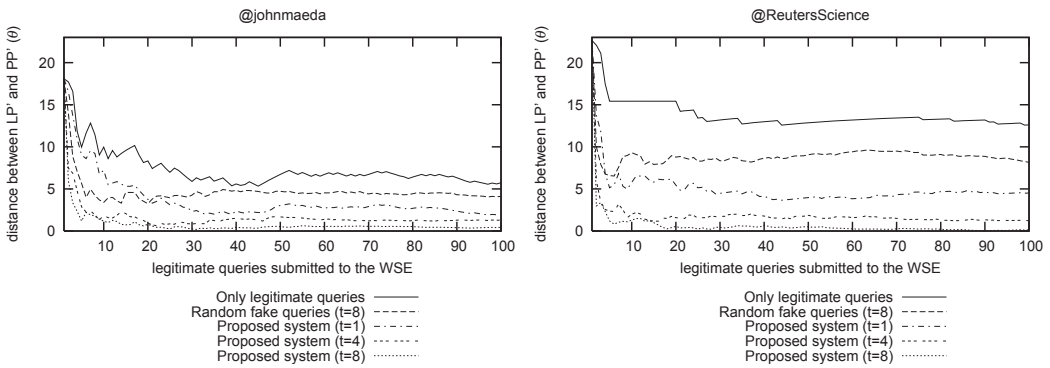


Fig. 1. Distance between profiles considering the same set of legitimate queries submitted by two different users

TABLE I
PP' BUILT FROM THE SET OF QUERIES ψ AND LP' BUILT FROM THE TWEETS PUBLISHED BY @JOHNMAEDA AND @REUTERSSCIENCE

Category	Arts	Health	Shopping	Science	Computers	Sports	Society	Business
$\mathbf{PP'_{\psi}}$	25%	8%	14%	3%	21%	3%	6%	20%
$\mathbf{LP'_{@johnmaeda}}$	27,6%	1,5%	4,5%	8,6%	14,1%	3,0%	19,1%	21,6%
$\mathbf{LP'_{@ReutersScience}}$	9,4%	2,2%	3,8%	42,2%	8,3%	2,5%	17,2%	14,4%

predominant focus on *Arts* and *Science* respectively. These users are @johnmaeda for *Arts* and @ReutersScience for *Science*. Both users were selected using the tool *WhoToFollow* provided by Twitter.

From each Twitter user, 100 tweets were extracted and profiled using the classifier introduced in Section II-A. The two different normalized local profiles LP' which were built are shown in Table I.

Considering that PP' and LP' are defined as $\{< c_1, \lambda_1 > , \ldots, < c_k, \lambda_k >\}$, we compute the distance between both profiles after each submitted query as follows:

$$\theta(LP', PP') = \frac{\sum_{i=1}^{k} |(\lambda_i; LP') - (\lambda_i; PP')|}{k}$$

When θ is close to 0, it means that there is no difference between LP' and PP'. On the other hand, the larger θ is, the bigger the difference between LP' and PP' becomes.

The proposed method is compared with the two following scenarios: (i) the two tested users submit only the queries in ψ; and (ii) they use a naive privacy-preserving method that adds t random *fake* queries extracted from ODP per each *legitimate* user query.

Figure 1 shows that, in all the cases, our approach clearly outperforms the random method, even when our system generates only one *fake* query per each *legitimate* one ($t = 1$) and the random approach generates eight *fake* queries ($t = 8$).

IV. CONCLUSIONS

In this paper, we have proposed a new scheme that distorts user profiles by submitting *fake* queries to the WSE. Our system locally profiles users according to their social network accounts (e.g., Twitter). This strategy is assumed to provide better personalized service than other schemes in the literature. The privacy dimension is addressed by building *fake* queries that keep the *macro-interests* of the user while hiding her *micro-interests*.

ACKNOWLEDGMENTS

This work was partly supported by the European Commission under FP7 project Inter-Trust, by the Spanish Ministry of Science and Innovation (through projects eAEGIS TSI2007-65406-C03-01, CO-PRIVACY TIN2011-27076-C03-01, ARES-CONSOLIDER INGENIO 2010 CSD2007-00004) and by the Government of Catalonia (2009 SGR 1135).

REFERENCES

[1] S. Hansell, "Increasingly, Internet's Data Trail Leads to Court", *The New York Times*, February, 2006.
[2] M. Barbaro and T. Zeller, "A Face Is Exposed for AOL Searcher No. 4417749", *The New York Times*, 2006.
[3] A. Viejo, J. Castellà-Roca, "Using Social Networks to Distort Users' Profiles Generated by Web Search Engines", *Computer Networks*, vol. 54, no. 9, pp. 1343–1357, 2010.
[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] D.C. Howe, H. Nissenbaum, "TrackMeNot: Resisting surveillance in web search", *Lessons from the Identity Trail: Anonymity, Privacy, and Identity in a Networked Society*, chapter 23, pp. 417-436. Oxford University Press, 2009.
[6] 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. 4, pp. 1468-4527, 2009.
[7] B. Shapira, Y. Elovici, A. Meshiach, T. Kuflik, "PRAW - A PRivAcy model for the Web", *Journal of the American Society for Information Science and Technology*, vol. 56, no. 2, pp. 159–172, 2005.
[8] D. Sánchez, J. Castellà-Roca, A. Viejo, "Knowledge-Based Scheme to Create Privacy-Preserving but Semantically-Related Queries for Web Search Engines", *Information Sciences*, vol. 218, pp. 17–30, 2013.
[9] A. Viejo, D. Sánchez, J. Castellà-Roca, "Using Profiling Techniques to Protect the User Privacy in Twitter", *Proc. of 9th International Conference on Modeling Decisions for Artificial Intelligence – MDAI'12*, 2012.
[10] K. Zoltan, S. Johann, "Semantic Analysis of Microposts for Efficient People to People Interactions", *Proc. of the Roedunet International Conference RoEduNet'11*. pp. 14, 2011.