Studies in Computational Intelligence 567

Guillermo Navarro-Arribas
Vicenç Torra *Editors*

Advanced Research in Data Privacy

Springer

Studies in Computational Intelligence

Volume 567

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

More information about this series at http://www.springer.com/series/7092

Guillermo Navarro-Arribas
Vicenç Torra
Editors

Advanced Research in Data Privacy

Springer

Editors
Guillermo Navarro-Arribas
Department of Information and Communications Engineering
Universitat Autonoma de Barcelona
Catalonia
Spain

Vicenç Torra
Institut d'Investigació en Intel·ligència Artificial
Consejo Superior de Investigaciones Científicas Campus de la UAB
Catalonia
Spain

ISSN 1860-949X
ISBN 978-3-319-09884-5
DOI 10.1007/978-3-319-09885-2

ISSN 1860-9503 (electronic)
ISBN 978-3-319-09885-2 (eBook)

Library of Congress Control Number: 2014947701

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This book presents the research work done under the auspices of the ARES Project (CSD2007-00004). ARES, which stands for Advanced Research in Privacy and Security, has been one of the most ambitious research projects on computer security and privacy funded by the Spanish Government. It is part of the now extinct CONSOLIDER INGENIO 2010 program, a highly competitive program that aimed to advance knowledge and open new research lines among top Spanish research groups.

The ARES project, coordinated by Josep Domingo-Ferrer from Universitat Rovira i Virgili, started in 2007 and was composed of six research groups from six different institutions: Universitat Rovira i Virgili, Consejo Superior de Investigaciones Científicas, Universidad de Málaga, Universitat Oberta de Catalunya, Universitat Politecnica de Barcelona, and Universitat de les Illes Balears. After 7 years, the project is about to conclude this September 2014. It has given important and internationally recognized results in the areas of computer security and privacy, has significantly increased the research production, and has fueled technology transfer activities.

Among the ARES project, privacy has played an important role. Our group led the work package about privacy within the project, for which Vicenç Torra was mainly responsible. Our intention with this book is to provide a guide to the research done within the ARES project in relation to privacy.

Participants of the project were invited to submit a chapter on their contribution to data privacy and privacy enhancing technologies. These submissions were handled through a peer-review process ending in the current chapters that form the book. In addition, there are three introductory chapters: one that introduces the book, and two introducing the work of the main groups on privacy within ARES. At least one author of each contribution is, or has been, in the ARES project.

This is not an exhaustive enumeration of the work done in the ARES project related to privacy. Instead of giving an exhaustive list we opted for allowing the contributors to actually choose the work that they feel was more relevant and interesting for future research. This book aims to introduce and spread the work

done in ARES directly related to privacy. We think it also serves as a review of the current research trends in privacy and privacy enhancing technologies.

We would like to thank all the authors and reviewers that have kindly contributed to this book, as well as Prof. J. Kacprzyk for his support on publishing this book, and the editorial team at Springer for their help.

Bellaterra, June 2014

Guillermo Navarro-Arribas
Vicenç Torra

Contents

Part I
Introduction

Advanced Research on Data Privacy in the ARES Project

Guillermo Navarro-Arribas and Vicenç Torra

Abstract Privacy has become an important concern in today's society. The advancement and pervasiveness of information and communication technologies have a great positive impact in our society, they greatly affect how we socialize, the way we do business, or even our individual and social freedom.

1 Introduction

Privacy has become an important concern in today's society. The advancement and pervasiveness of information and communication technologies have a great positive impact in our society, they greatly affect how we socialize, the way we do business, or even our individual and social freedom. At the same time, these new technologies are enabling an unparalleled invasion of privacy. There has been a relatively recent awareness regarding government mass surveillance programs [13], important information leakages in corporation environments [6], or even highly publicized scandals arousing when poorly anonymized user data is made public [4, 25]. Governments, users, and corporations are starting to take privacy seriously. As an example of user awareness, a recent survey from Mozilla identifies privacy as the top priority of users from all regions concerning the future of the Web [24]. All this interest has motivated an increased interest from the research community in data privacy and privacy

G. Navarro-Arribas (✉)
Department of Information and Communications Engineering,
Universitat Autonoma de Barcelona, 08193 Bellaterra, Catalonia, Spain
e-mail: gnavarro@deic.uab.cat

V. Torra
Institut d'Investigació en Intel·ligència Artificial,
Consejo Superior de Investigaciones Científicas Campus de la UAB,
08193 Bellaterra, Catalonia, Spain
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_1

enhancing technologies. From more traditional disclosure control techniques rooted in the statistics community to more recent studies in social networks or data mining. The increase in research work from private and public sectors has driven the raise of funded research projects and academic production.

A relevant academic project funded by the Spanish Government is about to conclude in September 2014. The project was called ARES, Advanced Research on Information Security and Privacy, and, as expected, an important part of the project was to advance the research on data privacy.

In this book we give an overall picture of the most notable research work that has come out from the ARES project in relation to data privacy. All the works presented here, have been done under the project umbrella. At the same time these works do provide an state of the art picture of data privacy research, since they include important contributions already recognized in prestigious international conferences and journals.

In the following section we describe the book contents to give an idea of what the reader will find in the rest of book.

2 The ARES Project and Data Privacy

The ARES project was part of the, currently extinguished, CONSOLIDER INGENIO 2010 program, possibly the most ambitious and competitive research program in Spain. The project is composed of six Spanish research groups in the area of information security and privacy. It started in October 2007 and will end in September 2014.

The specific groups that took part in the project are:

- CRISES/URV: Secure Electronic Commerce group at Universitat Rovira i Virgili of Tarragona.
- IF-PAD/CSIC: IF-PAD, Information Fusion for Privacy and Decision group, located at the Artificial Intelligence Research Institute of CSIC.
- KISON/UOC: K-ryptography and Information Security for Open Networks group from the Universitat Oberta de Catalunya.

Table 1 Groups and their principal investigators taking part in the ARES project

Group	P.I.	Web page
CRISES/URV	Josep Domingo-Ferrer	http://crises2-deim.urv.cat/
IF-PAD/CSIC	Vicenç Torra	http://www.iiia.csic.es/~vtorra/ares/
KISON/UOC	Jordi Herrera-Joancomartí, David Megias	http://kison.uoc.edu
ISG/UPC	Miguel Soriano	http://isg.upc.edu/
GIDET/UIB	Josep Lluís Ferrer-Gomila	http://secom.uib.es/
GSI/UMA	Javier Lopez	https://www.nics.uma.es/

- ISG/UPC: Information Security Group at the Universitat Politècnica de Catalunya, in Barcelona.
- GIDET/UIB: the Interdisciplinary Group on Law and Telematics at the Universitat de les Illes Balears.
- GSI/UMA: the Network Information and Computer Security Lab (NICS), former Information Security Group (GSI), from the Universidad de Málaga.

A summary of the groups, the principal investigator, and web page for each group is given in Table 1. The project coordinator has been Josep Domingo-Ferrer from the CRISES/URV group.

The aim of the project was to create technologies to conciliate security, privacy and functionality in the information society.

More precisely, the research work of the project has been settle around three intertwined applications scenarios plus two transversal underpinning areas:

- Scenario 1: protection of critical information infrastructures.
- Scenario 2: ubiquitous computing.
- Scenario 3: secure electronic commerce and digital content distribution.
- Underpinning area 1: data privacy technologies.
- Underpinning area 2: technical-legal issues.

Each gives up to a specific workpackage, two of them transversal: data privacy technologies and technical-legal issues. Two more workpackages are intended for management and for field trial and dissemination. Table 2 lists the workpackages with the group that led each one, and Fig. 1 summarizes the workpackages and their interrelation. A links from WPi to WPj means that work done in WPi is used by WPj.

As shown, research on privacy is gathered in an specific work package within the project. The WP4, Data privacy technologies, is a transversal worpackage which has been leaded by the IF-PAD/CSIC group. Its main (broad) objective has been to develop or adapt privacy technologies to solve privacy problems aroused form other parts of the projects.

As we will see, the work presented through the book develops several topics and provides an overview of the research carried by the project participants. Some of the presented works include collaborations with researchers outside the project.

Table 2 Workpackages of the ARES project with its leader group

WP1	Critical infrastructure protection	GSI/UMA
WP2	Ubiquitous computing	ISG/UPC
WP3	Secure e-commerce and digital content distribution	KISON/UOC
WP4	Data privacy technologies	IF-PAD/CSIC
WP5	Technical-legal issues	GIDET/UIB
WP6	Field trial, technology transfer, and dissemination	CRISES/URV
WP7	Management	CRISES/URV

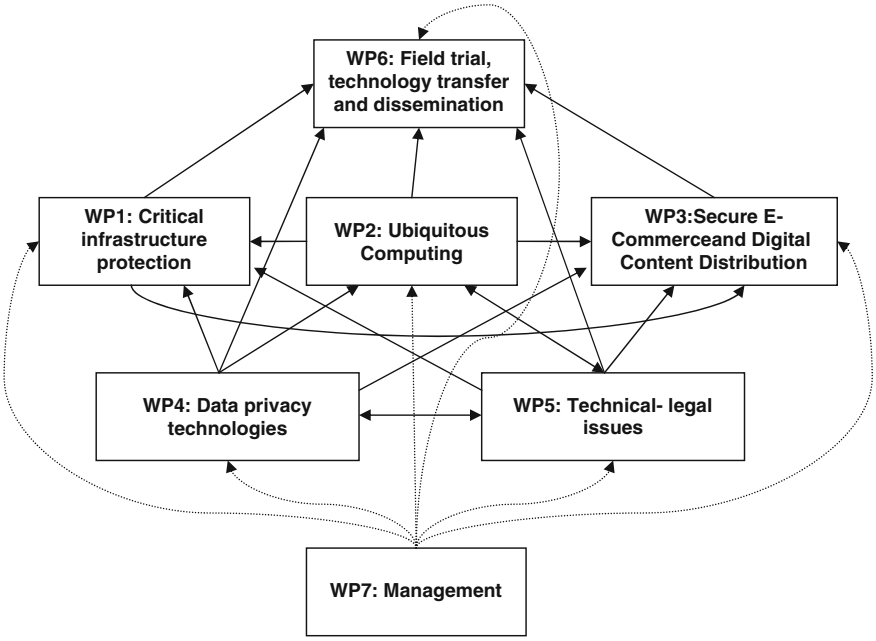


Fig. 1 ARES workpackages

3 Data Privacy

Data privacy and privacy enhancing technologies (PET) provide a broad research line which spreads through several specific fields. The common aim is to study the protection of information in order to avoid unintentional disclosure of sensitive information. One of the first disciplines to deal with this problem is Statistical Disclosure Control (SDC). SDC, rooted in the statistics community, develops methods to allow the publication of data from statistical agencies while preserving the privacy of their users. In the case of computer science, data privacy has become an important research line. Under privacy enhancing technologies we find from cryptography to privacy-preserving data mining or private information protocols, …Although this comprises several disciplines all share the same goal and even some of the techniques, methods and definitions.

The classification of privacy enhancing technologies is difficult due to the overlap in most of the disciplines. Nevertheless a well accepted dimension to classify them is determined by considering whose privacy is being sought [10]. That is, techniques can be classified in terms of whose privacy are attempting to protect. We have thus:

- **Respondent privacy** The respondent is the subject to whom the data is referring to. In the context of SDC, for example in a Census database, the respondents are the concrete subjects included in the Census. The respondent is considered a passive subject, who cannot act within the system to protect its own data.

- **Owner privacy** The owner is the proprietary or administrator holder of the data. It is normally the one liable for disclosure of sensitive information.
- **User privacy** The user can be seen as the active counterpart of a respondent. That is, the user is a subject that can actively participate in the protection of its own private data. This is usually done through the interaction of the subject with the system.

We have used the term subject to denote any entity taking part in the system. Although subjects will usually be individuals, other types of entities can be accommodated in the previous dimension. Examples of subjects can be: human beings, organizations, computer processes, electronic devices, …

As we will see all the works presented in this book are either respondent or user privacy approaches. Owner privacy, although important, is rarely found in common scenarios.

Besides respondent, owner, and user, other classification can be found in the literature [2, 11, 12, 16, 26, 35, 37, 40, 41]. For instance it is common to differentiate data privacy methods by their intended use, or by the source of the data to be protected. We have however opted for the already mentioned classification. As we will see, in some cases the classification of an specific work into a concrete class is somewhat fuzzy, since some methods and technologies can address the protection of more than one type of entity.

In the following sections we describe the chapters of the book organized according to the previous classification.

Together with this introduction, there are two introductory chapters that summarize the work carried out by two of the groups of the project. These are the groups with a stronger presence in privacy technologies within the ARES project.

Manjón and Domingo-Ferrer [18] (Chap. 2) describe the work carried out by the group from the Universitat Rovira i Virgili within the ARES project. This group and his leader, Josep Domingo-Ferrer, have been the main coordinator of the ARES project, with a strong participation in the whole project. This chapter allows the reader to get a picture of the work performed by the URV group regarding privacy technologies. Note that some of the works described involve the participation of other groups from the project, and external collaborators.

Torra and Navarro-Arribas [36] (Chap. 3) describe the work done by the IIIA-CSIC group regarding data privacy within the ARES project. This group and his leader Vicenç Torra, have been in charge of leading the workpackage WP4 on data privacy technologies (see Sect. 2).

4 Respondent Privacy

Respondent privacy is possibly the most common case found in data privacy scenarios. Most common Statistical Disclosure Control (SDC) and Privacy-preserving Data Mining (PPDM) techniques usually fit in this category. We present in this book several state of the art works concerned with respondent privacy in several application scenarios.

4.1 SDC and PPDM

Statistical Disclosure Control and Privacy-preserving Data Mining are possibly the most classical works on data privacy. Their common objective is to protect a dataset so statistical analysis and data mining techniques can be applied while preserving the privacy of the respondents.

Matwin et al. [23] (Chap. 4) provide an introductory review of data privacy from statistical disclosure control (SDC) and privacy-preserving data mining (PPDM). Authors make specific emphasis in attribute disclosure control.

Abril et al. [1] (Chap. 5) provide an introduction to privacy-preserving data mining (PPDM) with R. R has become an important language and environment for statistics and data mining and thus it is very well suited for SDC and PPDM. This chapter serves as an introduction to PPDM protection techniques, and information loss and disclosure risk, outlining tools and procedures in R to help introducing practitioners to this field.

Marés et al. [19] (Chap. 6) study the problem of finding optimal transition matrices for Post Randomization Methods (PRAM). PRAM is a method commonly used in SDC to introduce perturbation by using a Markov probability transition matrix. The authors introduce a method based in genetic algorithms to find the optimal matrix. That is, the one with a better balance between disclosure risk and utility.

4.2 Semantic Related Respondent Privacy Protection

The following chapters, depart from the traditional approach of SDC and PPDM to deal with textual data. Traditional SDC and PPDM methods usually deal with numeric or categorical data. In the later case they rely at most in a predefined generalization tree. Recent work has been made to deal with textual data by considering its semantics in a broad sense. This allows to deal with free text, or categorical data without predefined categories. These three chapters show the use of semantic based protection techniques and also introduce the problem of document sanitization.

Martínez et al. [20] (Chap. 7) consider the anonymization of categorical datasets using semantic information. The authors consider well known anonymization methods from SDC such as recoding, microaggregation, and resamplig. These methods are then adapted to take into account the semantics of the data they are protecting, usually relaying in ontologies to model the semantic knowledge associated with the attributes of the dataset.

Batet and Sanchez [5] (Chap. 8) go in depth with semantic privacy methods by reviewing semantic similarity functions. Several SDC and PPDM methods such as microaggregation, additive noise, recoding, sampling, or data swapping, require to some extend the use of a distance or similarity function. The chapter serves as a survey of semantic similarity functions to be used in such privacy protection methods.

Nettleton and Abril [28] (Chap. 9) tackle the problem of document sanitization. The sanitization process allows to disclose a confidential document by removing, generalizing, or distorting the confidential information contained in the document. The authors evaluate the sanitization process using information retrieval metrics.

4.3 Location Privacy

A very specific type of data that has gained recent popularity is the one related to localization. The advances in localization technology have made it very easy to collect location data from smartphones, GPS devices,…. The possibility of mining these data opens up interesting application, but at the same time expose the privacy of the respondents of the data. Here we will see three approaches to deal with location data, two of them treat trajectory data, which consider location and timing (e.g. vehicle trajectories within a urban environment).

Conti et al. [8] (Chap. 10) review user privacy in location based services based on footprints. A footprint considers the amount of time that the user spends in a given area. The authors show the risk and weakness found in this type of anonymization models when facing an adversary with previous knowledge not considered by the anonymization procedure. This analysis leads the authors to conclude with a set of properties to determine the actual level of privacy of these models. In this scenarios the actual anonymization is not performed by the users as active subjects but by a trusted server, the so called location depersonalization server. It is this service who also discloses the protected data regarding the users.

Trujillo-Rasua et al. [38] (Chap. 11) depict a review of privacy methods for spatio-temporal databases. More precisely, authors provide a review of microaggregation to protect data related to movement, or trajectories. A trajectory can be seen as a location data with a temporal component. The chapter is concluded with a concrete proposal and evaluation of an specific microaggregation method for trajectories.

Martínez-Bea [22] (Chap. 12) also considers the anonymization of data describing trajectories using microaggregation. In this case, the protection mechanism is based on time series. That is, the proposal departs from previous work on the anonymization of time series and applies it to trajectories. As the chapter describes, this method was implemented as part of a demonstrator of the ARES project [3].

5 Social Networks

Social networks, by their intrinsic nature, expose sensitive information from their users. Protecting the privacy of users in social networks is a hot research topic. When we consider the respondent privacy approaches in social networks, we are assuming that the network authority or a trusted third party performs the anonymization. In this case we will also see metrics to measure privacy.

Casas-Roma et al. [7] (Chap. 13) consider the protection of graph data. These data usually corresponds to social network relationships, which can be considered as sensitive information. The authors review privacy preserving methods for networks based on the k-anonymity property. The chapter includes an empirical evaluation of the methods.

Sramka [33] (Chap. 14) provides a review of privacy metrics in the context of social networks. Furthermore, the author introduces a novel privacy metric. These metrics are very useful in order to assess the privacy exposure of the users of a social network. Users can be aware of how their sensitive information is being distributed in the network. We have classified this work as respondent privacy since the computation of the proposed metric requires the consideration of the whole network. Something that, in some cases, is only available to the social network administrative authority and not individual users.

5.1 Other Respondent Privacy Enhancing Technologies

The consideration of privacy in other systems to protect the anonymity of the respondents is also increasing in recent years. Privacy is being considered in authentication schemes, electronic ticketing systems and coupons booklets, in RFID technology, or in video surveillance.

Martínez-Ballester et al. [21] (Chap. 15) present a review of Trustworthy Video Surveillance System (T-VSS). Their work faces the problem of anonymizing surveillance video files to mitigate the disclosure of individual personal data. The authors present a privacy-aware video surveillance platform that can be used as a safety protection while preserving the privacy of individuals.

Payeras-Capellà et al. [30] (Chap. 16) introduce the study of privacy issues in electronic ticketing systems. They analyze the requirements and state of the art in electronic tickets used in transport services, and highlight required properties regarding the privacy and anonymity of users.

Garcia-Alfaro et al. [14] (Chap. 17) consider privacy issues related to passive RFID tags. More precisely the authors introduce and analyze the EPC Gen2 technology identifying security and privacy threats. The authors also survey countermeasures applicable to mitigate the identified threats. The chapter also outlines the work done within the ARES project in relation to this topic and discuss future research directions.

Hinarejos et al. [15] (Chap. 18) consider electronic coupons booklets. The authors review the state of the art of these systems which are the electronic equivalent of paper coupons booklets, usually offered as discount tickets to users. The chapter outlines the security and privacy requirements of these systems, and propose a solution for mobile scenarios.

Daza and Signorini [9] (Chap. 19) review authentication technologies taking into special consideration its use as a privacy enhancing technology. Moreover the authors discuss hardware intrinsic security (HIS) approaches and present the APtItUDe system which while using physically unclonable functions, avoids the use of challenge-response databases. This system guarantees a high level of data privacy while providing a user friendly authentication process.

6 User Privacy

This other part of the book deals with the user privacy approaches. We will see several systems and methods where the users perform active actions to ensure the protection of their privacy. As an example of user privacy we will consider how users can actively protect their privacy against a web search engine, or recommender and personalized information systems.

6.1 Web Search Engines

The information gathered by a Web search engine (WSE) can undoubtedly raise important privacy concerns to their users. There are two approaches to allow users to use a WSE without reveling their sensitive information. The first one is when the users trust the WSE to perform an anonymization procedure on the gathered data [27, 31]. This will be the case of a respondent privacy approach to protect WSE information. The second approach, on the contrary, relies on the users themselves to perform the required actions to ensure their own privacy. We will see here examples of this second approach.

Romero-Tris et al. [32] (Chap. 20) review the so called multi-party approach for user anonymization of queries. These kind of methods rely in multi-party protocols performed by a group of cooperative users in order to hide the real preferences of each individual user within the group. The authors also propose some improvements over the Useless User Profile protocol, which allow users to security exchange their queries. This allows each user to submit a query from her partners distorting the profile that the WSE can build for her.

Juarez and Torra [17] (Chap. 21) also deal with the problem of anonymizing user profiles to a WSE. They discuss DisPA (Dissociating Privacy Agent), a browser extension, which allows the user to increase its privacy against a WSE. To do that, DisPA semantically disassociates search queries by topics, which are sent with different profiles. To the eyes of the WSE these are queries coming from different users, but given that they are semantically grouped, they allow certain degree of personalization by the WSE.

Stokes and Bras-Amorós [34] (Chap. 22) introduce the use of combinatorial configurations to model peer-to-peer private information retrieval protocols. Although it refers to UPIR protocols, it can be contextualized also in terms of WSE, and more precisely as multi-party methods from [32] or the untrusted model discussed in [29]. The users collaborate to perform the query in a database. The authors provide a review and introduce important concepts and approaches to deal specifically with user collusion and anonymous neighbors.

6.2 Recommender and Personalized Systems

Recommender and personalized systems are becoming very important. Contrary to what we have seen regarding social networks (see Sect. 5), here we will show user privacy methods in a very similar context. In these cases it is the user who, to some extent, takes action to protect its own sensitive information.

Parra-Arnau et al. [29] (Chap. 23) provide a review of privacy in personalized information systems. The authors review the state of the art and classify existing proposals. At the same time they also review privacy metrics for this personalized information systems. In their review, the authors distinguish between a *trusted* model, which requires a trusted third party to perform the anonymization, *untrusted* model, where there is no trusted party and the anonymization procedure relies in the users, and *semi-trusted* model, when the users collaborate among peers to perform the anonymization (in the same line as the multi-party methods for queries presented in [32]). Regarding our broad classification, the trusted model will yield respondent privacy methods, while untrusted and semi-trusted models correspond to user privacy. We have included this work as a user privacy approach since we feel it can be more interesting from this point of view.

Vera del Campo et al. [39] (Chap. 24) address the problem of privacy in recomendation systems. The work is contextualized in the Internet of Things, and presents DocCloud. DocCloud is a document recommender system, which provides several privacy-related protections, which here is extended to generic cloud resources in the context of a social network. We have classified this work as user privacy, although the anonymization is somehow ensured by the infrastructure.

7 Conclusions

This chapter introduces the current book on data privacy. Although this is not an exhaustive enumeration, the book gives a broad picture of the work done under the umbrella of the ARES research project.

The chapters of the book are contextualized in respondent and user privacy. The chapters present state of the art research in several fields such as statistical disclosure control and privacy-preserving data mining, security technologies, and social networks.

Acknowledgments Partial support by the Spanish MICINN (projects COPRIVACY (TIN2011-27076-C03-03), N-KHRONOUS (TIN2010-15764), and ARES (CONSOLIDER INGENIO 2010 CSD2007-00004)) and by the EC (FP7/2007-2013) Data without Boundaries (grant agreement number 262608) is acknowledged.

References

1. Abril, D., Navarro-Arribas, G., Torra, V.: Data privacy with R. Chapter 17, Advanced Research on Data Privacy. Springer, Cham (2014)
2. Aggarwal, C.C., Yu, P.S.: A general survey of privacy-preserving data mining models and algorithms. Privacy-Preserving Data Mining, Advances in Database Systems, pp. 11–52. Springer, New York (2008)
3. Aragonés, J., Manjón, J.A.: Field trial for joint validation and media dissemination of WP1-WP2-WP3-WP4 technologies in a real-world vehicular network environment (WP6.T1) Deliverable Report. ARES project CONSOLIDER-INGENIO 2010 CSD2007-00004 (2012)
4. Barbaro, M., Zeller, T.: A Face is Exposed for AOL Searcher No. 4417749. The New York Times, New York (2006). Accessed 9 Aug 2006 (Accessed 25 Apr 2010)
5. Batet, M., Sanchez, D.: Contributions on Semantic Similarity and its Applications to Data Privacy. Chapter 18, Advanced Research on Data Privacy. Springer, Cham (2014)
6. BBC News: Sony faces legal action over attack on PlayStation network. BBC news technology. http://www.bbc.co.uk/news/technology-13192359 (2011). Accessed 28 Apr 2011
7. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: A Summary of k-Degree Anonymous Methods for Privacy-Preserving on Networks. Chapter 13, Advanced Research on Data Privacy. Springer, Cham (2014)
8. Conti, M., Di Pietro, R., Marconi, L.: Privacy for LBSs: on Using a Footprint Model to Face the Enemy. Chapter 10, Advanced Research on Data Privacy. Springer, Cham (2014)
9. Daza, V., Signorini, M.: Smart User Authentication for an Improved Data Privacy. Chapter 19, Advanced Research on Data Privacy. Springer, Cham (2014)
10. Domingo-Ferrer, J.: A Three-Dimensional Conceptual Framework for Database Privacy, Secure Data Management. Lecture Notes in Computer Science, pp. 193–202. Springer, Berlin Heidelberg (2007)
11. Domingo-Ferrer, J., Torra, V.: Disclosure control methods and information loss for microdata. In: Doyle, P., Lane, J.I., Theeuwes, J.J.M., Zayatz, L. (eds.) Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 91–110. Elsevier Science, Amsterdam (2001)
12. Duncan, G.T., Elliot, M., Salazar, J.J.: Statistical Confidentiality. Springer, New York (2011)
13. EFF: NSA Spying on Americans. Electronic frontier foundation. https://www.eff.org/nsa-spying (2014)
14. Garcia-Alfaro, J., Herrera-Joancomartí, J., Melià-Segui, J.: Security and Privacy Concerns about the RFID layer of EPC Gen2 Networks. Chapter 19, Advanced Research on Data Privacy. Springer, Cham (2014)
15. Hinarejos, M.F., Isern-Deyà, A.P., Ferrer-Gomila, J.L.: Privacy on Mobile Coupons Booklets. Chapter 20, Advanced Research on Data Privacy. Springer, Cham (2014)
16. Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Nordholt, E.S., Spicer, K., de Wolf, P.-P.: Statistical Disclosure Control. Wiley, New York (2012)
17. Juarez, M., Torra, V.: Optimisation-Based Study of Data Privacy by Using PRAM. Chapter 21, Advanced Research on Data Privacy. Springer, Cham (2014)
18. Manjón, J.A., Domingo-Ferrer, J.: Selected Privacy Research Topics in the ARES Project: An Overview. Chapter 2, Advanced Research on Data Privacy. Springer, Cham (2014)
19. Marés, J., Torra, V., Shlomo, N.: Optimisation-Based Study of Data Privacy by Using PRAM. Chapter 6, Advanced Research on Data Privacy. Springer, Cham (2014)
20. Martínez, S., Valls, A., Sanchez, D.: Semantic Anonymisation of Categorical Datasets. Chapter 7, Advanced Research on Data Privacy. Springer, Cham (2014)
21. Martínez-Balleste, A., Solanas, A., Rashwan, H.A.: Trustworthy Video Surveillance: an Approach Based on Guaranteeing Data Privacy. Chapter 15, Advanced Research on Data Privacy. Springer, Cham (2014)
22. Martinez-Bea, S.: A Prototype for Anonymizing Trajectories from a Time Series Perspective. Chapter 12, Advanced Research on Data Privacy. Springer, Cham (2014)

23. Matwin, S., Nin, J., Sehatkar, M., Szapiro, T.: A Review of Attribute Disclosure Control. Chapter 4, Advanced Research on Data Privacy. Springer, Cham (2014)
24. Mozilla: The web we want. https://webwewant.mozilla.org/ (2014)
25. Narayanan, A., Shmatikov, V.: Robust De-anonymization of Large Sparse Datasets. In Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP '08), pp. 111–125. IEEE Computer Society (2008)
26. Navarro-Arribas, G., Torra, V.: Information fusion in data privacy: a survey. Inf. Fusion **13**(4), 235–244 (2012)
27. Navarro-Arribas, G., Torra, V., Erola, A., Castellà-Roca, J.: User k-anonymity for privacy preserving data mining of query logs. Inf. Process. Manage. **48**, 476–487 (2012)
28. Nettleton, D.F., Abril, D.: An Information Retrieval Approach to Document Sanitization. Chapter 9, Advanced Research on Data Privacy. Springer, Cham (2014)
29. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: Privacy-Enhancing Technologies and Metrics in Personalized Information Systems. Chapter 23, Advanced Research on Data Privacy. Springer, Cham (2014)
30. Payeras-Capellà, M.M., Mut-Puigserver, M., Ferrer-Gomila, J.L., Castellà-Roca, J., Vives-Guasch, A.: Electronic Ticketing: Requirements and Proposals Related to Transport. Chapter 16, Advanced Research on Data Privacy. Springer, Cham (2014)
31. Poblete, B., Spiliopoulou, M., Baeza-Yates, R.: Privacy-preserving query log mining for business confidentiality protection. ACM Trans. Web **4**(3), 1–26 (2010)
32. Romero-Tris, C., Viejo, A., Castellà-Roca, J.: Multi-party Methods for Privacy-Preserving Web Search: Survey and Contributions. Chapter 20, Advanced Research on Data Privacy. Springer, Cham (2014)
33. Sramka, M.: Evaluating Privacy Risks in Social Networks from the Users Perspective. Chapter 14, Advanced Research on Data Privacy. Springer, Cham (2014)
34. Stokes, K., Bras-Amorós, M.: A Survey on the Use of Combinatorial Configurations for Anonymous Retrieval. Chapter 22, Advanced Research on Data Privacy. Springer, Cham (2014)
35. Torra, V.: Towards Knowledge Intensive Data Privacy. Data Privacy Management and Autonomous Spontaneous Security. Lecture Notes in Computer Science, vol. 6514, pp. 1–7. Springer, Cham (2011)
36. Torra, V., Navarro-Arribas, G.: Data Privacy at the IIIA-CSIC. Chapter 3, Advanced Research on Data Privacy. Springer, Cham (2014)
37. Torra, V., Navarro-Arribas, G.: Data privacy. WIREs Data Min. Knowl. Discov. **4**, 178–195 (2014). doi:10.1002/widm.1129
38. Trujillo-Rasua, R., Domingo-Ferrer, J.: Privacy in Spatio-Temporal Databases: A Microaggregation-Based Approach. Chapter 11, Advanced Research on Data Privacy. Springer, Cham (2014)
39. Vera del Campo, J., Pegueroles, J., Hernandez-Serrano, J., Soriano, M.: Managing Privacy in the Internet of Things: DocCloud, a Use Case. Chapter 24, Advanced Research on Data Privacy. Springer, Cham (2014)
40. Verykios, V.S., Bertino, E., Fovino, I.N., Provenza, L.P., Saygin, Y., Theodoridis, Y.: State-of-the-art in privacy preserving data mining. SIGMOD Rec. **33**, 50–57 (2004)
41. Willenborg, L., de Waal, T.: Elements of Statistical Disclosure Control. Lecture Notes in Statistics. Springer-Verlag, New-York (2001)

Selected Privacy Research Topics in the ARES Project: An Overview

Jesús A. Manjón and Josep Domingo-Ferrer

Abstract This chapter gives an overview of some of the data privacy research carried out by the team at Universitat Rovira i Virgili within the ARES project. Topics reviewed include query profile privacy, location privacy, differential privacy and anti-discrimination.

1 Introduction

Data privacy is the adaptation to the Information Society of the fundamental right to privacy and private life, included by the United Nations in the Universal Declaration of Human Rights (1948), whose Article 12 states: "No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honour and reputation. Everyone has the right to the protection of the law against such interference or attacks".

The rise of information technologies has arisen new threats against personal privacy such as profiling, location tracking or reidentification. Data privacy technologies are about technically enforcing the above right in the information society.

In this chapter we give a general overview of the data privacy research carried out by the team at Universitat Rovira i Virgili. The topics covered here relate to privacy in databases: user privacy (i.e. query profile privacy), respondent privacy (i.e. data anonymization) and anti-discrimination protection in data mining.

J.A. Manjón (✉) · J. Domingo-Ferrer
Department of Computer Engineering and Mathematics,
Universitat Rovira I Virgili, UNESCO Chair in Data Privacy,
Av. Països Catalans 26, 43007 Tarragona, Catalonia, Spain
e-mail: jesus.manjon@urv.cat

J. Domingo-Ferrer
e-mail: josep.domingo@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_2

Section 2 deals with query profile privacy, Sect. 3 covers location privacy in location-based services. Work to increase the utility of data sets anonymized under the differential privacy model is reported in Sect. 4. Research on methods to protect against discrimination in data mining are covered in Sect. 5. Finally, conclusions are drawn in Sect. 6.

2 Query Profile Privacy

Data bases and web search engines (WSE, e.g. Google, Bing, etc.) are widely used to find a certain piece of data among a huge amount of information in the shortest possible time. However, these useful tools also pose a privacy threat to the users: database servers and WSE may profile their users by storing and analyzing past searches submitted by them. To address this privacy threat, several solutions have been proposed in the literature.

Private information retrieval (PIR) is possibly the most ambitious solution, but it falls short of being practically deployable. In PIR, a user wants to retrieve an item from a database or search engine without the latter learning which item the user is interested in. PIR was invented in 1995 by Chor et al. [5, 7] with the assumption that there are at least two copies of the same database, which do not communicate with each other. In the PIR literature, the database is usually modeled as a vector and it is assumed that the user knows the physical address of the sought item. Keyword PIR [6] is a more flexible form of PIR: the user can submit a query consisting of a keyword and no modification in the structure of the database is needed.

However, if one wishes to run PIR against a search engine, there are some fundamental shortcomings: (i) the server has no motivation to co-operate in the PIR protocol; (ii) it is not realistic to model a database (let alone the world-wide web) as a vector in which the user can be assumed to know the physical location of the keyword sought. Even keyword PIR does not really fit, as it still assumes a mapping between individual keywords and physical addresses (in fact, each keyword is used as an alias of a physical address). A WSE allowing only searches of individual keywords stored in this way would be much more limited than real engines like Google or Yahoo.

In the sequel, we give solutions that relax the property of PIR that the database or WSE should not know the item retrieved by the user. We restrict our ambitions to allowing the user to keep her query profile (query history) confidential. We start with standalone solutions, in which a user protects himself with no one else's help, and then we review P2P solution, in which users help each other to protect their query privacy.

2.1 Standalone Solutions

Domingo-Ferrer et al. [15] defines $h(k)$-private information retrieval ($h(k)$-PIR) as a practical compromise between computational efficiency and privacy. They also present $h(k)$-PIR protocols that can be used to query any database, which does not

even need to know that the user is trying to preserve his or her privacy. The proposed methods protect the privacy of user queries by adding fake keywords to the keywords really being searched by the user; to prevent the WSE from distinguishing the real from the fake keywords, the latter must be chosen as having a similar frequency of appearance as the former. As a result, the WSE is unable to unequivocally determine the real interests of their users. The quality of the results decreases with the increase in privacy (i.e. with the number of fake keywords being added), but the trade-off being obtained is excellent.

A prototype called GooPIR was developed in Java JDK 6.0 Standard Edition to implement this scheme (http://crises2-deim.urv.cat/technology/get/id/1). The prototype accepts queries consisting of single keywords and queries consisting of a logical AND of several keywords (with the limitation that independence between the keywords must be a plausible assumption). GooPIR locally masks the target keyword(s), submits the masked query to the Google search engine and then locally filters the results relevant to the target keyword(s).

A standalone solution that was developed in parallel with $h(k)$-PIR by researchers not in ARES is TrackMeNot [23]. Here, instead of adding fake keywords to the real keyword sought by the user, the user's real queries are left unaltered but the system keeps generating and submitting additional fake queries that the WSE cannot easily distinguish from the real ones.

2.2 User-Private Information Retrieval (UPIR)

Like [10, 11, 15, 23] propose to relax strict PIR in order to obtain a practical system. However, rather than altering the user's query with fake queries or cloaking the user's query in a set of queries in a standalone fashion, the user's query history is blurred with the help of a peer-to-peer user community: a user gets her queries submitted on her behalf by other users in the P2P community. In this way, the database still learns which item is being retrieved (which deviates from strict PIR), but it cannot obtain the real query histories of users, which become diffused among the peer users. We name the resulting PIR relaxation user-private information retrieval (UPIR). This approach certainly requires the availability of peers, not needed in standalone systems [15, 23], but it has some advantages: unlike [15], it does not require knowledge of the frequencies of all possible keywords and phrases that can be queried; unlike [23], it avoids the overhead of ghost query submission.

Note that what we offer is different from what can be achieved using anonymization systems based on onion routing, like Tor [44]. In an onion routing system, the transport of data is protected by bouncing the communication between a user and a server around a distributed network of volunteer relays, with a view to protecting against traffic analysis. However, such systems give no end-to-end protection (at the application level). Specifically, as long as a search engine (or a database server) can link the successive queries submitted by the same user (e.g. by using cookies or some other mechanism), the profiling and the re-identification capabilities of the search

engine are unaffected even if the user is submitting her queries through Tor[1]: the user still submits all of her queries herself (the relays merely relay them), so her query history is unaltered and a query history may suffice for re-identification (see discussion in Sect. 2.5).

What [10, 11] propose is to diffuse a user's query profile among the peers in a peer-to-peer community. However, onion routing systems can indeed complement our solution and be used for peers to communicate among themselves and hide their identity from each other at the transport level.

The new scheme uses a type of combinatorial design called configuration to manage the keys used by peers to communicate their queries to each other and reduce the number of required keys (see [25, 34] for background on designs and configurations). The use of configurations in cryptographic key management was not new (e.g. see [25]), but their use in private information retrieval was.

2.3 Combinatorial Configurations

As indicated in the previous section, configurations are a combinatorial structure playing a central role in UPIR systems. We have done some research on configurations to improve our UPIR protocols.

A first contribution on configurations was [3], followed by [37]. In this latter paper it was proven that the optimal configurations for the P2P UPIR protocol presented in [10, 11] are the finite projective planes. This paper also presented an efficient and explicit algorithm for constructing finite projective planes. Finally, another aspect on the optimality of finite projective planes was treated: a short proof that they are Ramanujan graphs was given.

Subsequent contributions on configurations that were produced in ARES include [4, 38, 39].

2.4 Other Collaborative Solutions for Query Profile Privacy

In Castellà-Roca et al. [9] a collaborative approach was presented in which a central node groups users and collects the queries that users want to submit. Then the users execute an anonymous query retrieval protocol and each user obtains from the central node one query without knowing whose query it is. The user submits the query and broadcasts the WSE answer to all other users. This system had the shortcoming that the answer is made public and it might be linkable to the user who originated the query (e.g. in case of vanity query this is clear).

[1] However, using the Torbutton browser add-on helps eliminating cookings and allows using Tor to protect the query history of a user.

In [41] a social network was used for the first time. A new scheme was proposed that was designed to protect the privacy of the users from a web search engine that tries to profile them. The system provides a distorted user profile to the web search engine, because some of each user's queries are submitted by his/her friends in the social network. The proposed protocol submits standard queries to the web search engine, so it does not require any change on the server side. In addition to that, this scheme does not require the server to collaborate with the users.

Nevertheless, the following research questions appear when considering this scheme:

- The privacy level achieved by the users of this proposal depends on the function that calculates the probability of submitting a query. Can this function be re-designed to improve the current results?
- Mechanisms to measure the privacy level achieved by the users are needed in order to compare different proposals. Is there a standard measure that can be used for this purpose?
- The simulations which are shown in [41] have been performed using synthetic queries (queries which are generated at random by a computer). Would the use of real queries (queries generated by humans) influence the behavior of this scheme in terms of privacy protection?

Erola et al. [19] addressed the above research questions:

- The function used to decide which user must submit a certain query to the WSE was studied and re-designed. As a result, the privacy level achieved by the users was improved.
- A new measure to estimate the privacy achieved by the users, the *Profile Exposure Level* (*PEL*s), was proposed.
- The tests were performed using real data extracted from the well-known AOL file [42]. In this way, the correct behavior of the proposed system was tested with queries which have been generated by real users.

These changes improved the privacy achieved by the users in the previous version, while preserving usability.

Previous proposals of privacy-preserving web search protocols significantly increased the query delay. This is the time that the users need to wait in order to obtain the results of their queries. For this reason, the protocol presented in [29] focused on reducing the query delay. The resulting scheme was implemented and tested in an open environment and the results showed that it achieves the lowest query delay which had been reported in the literature. On the other hand, the work presented in [28] focuses on improving the level of security of previous proposals. More specifically, this work proposed a multi-party protocol that protected the privacy of the user not only in front of the web search engine, but also in front of other members of her own group. The results showed that this scheme outperforms similar proposals in terms of computation and communication.

Castellà et al. [8] was developed in collaboration with the Distributed Computation Group of the University of Lleida. This work focused on the development of a P2P

network that groups users according to their search preferences. Once the users are classified, they execute a protocol that protects their privacy in front of the web search engine.

2.5 Query Log Anonymization

The search logs generated by a web search engine 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 were generated [24]. There is at least one well-known case of released search logs with poor anonymization, which turned out to reveal enough information to re-identify some users. The release was done by AOL in an attempt to help the information retrieval research community, and ended up not only in important damage to the privacy of AOL users, but also in a major damage to AOL itself with several class actions suits and complaints against the company [42]. Ideally, the search logs should be properly anonymized before they become public. The problem is that achieving an acceptable degree of privacy in search logs is not easy, as there is a trade-off between privacy and the usefulness of the data.

In [17] we presented a method for anonymizing query logs, so as to be able to make them publicly available without encroaching on the privacy of the users who issued the logged queries. To that end, we followed the same ideas found in statistical disclosure control, and proposed a novel microaggregation method to anonymize query logs. This approach ensures a high degree of privacy, and offers k-anonymity at the user level, while preserving some of the data usefulness. Moreover, and unlike most of the previous work, our approach took into account the semantics of the queries in the anonymization process; this was achieved by using the Open Directory Project [43] ontology when aggregating the queries. A more extended version was presented in [18].

Another approach to microaggregating query logs was presented in [26]. In this paper, we defined a new user distance and an aggregation operator. The user aggregation was designed in order to be as computationally efficient as possible. Note that the most important part is the aggregation of the queries, since it is the information that will be most valuable in future analyses. Note also that queries are aggregated separately. An alternative could be to actually mix the terms of queries from different users and end up with new queries that somehow summarize all the users' queries. We chose the former approach given the complexity of the latter, and also because the former method yielded already satisfactory results.

As usual in statistical disclosure control techniques, there is a trade-off between privacy and usability. We showed that our proposals, besides providing k-anonymity, preserve to a good extent the information of the original logs. Our proposals can be regarded as an efficient and relatively simple method to protect query logs, and they ensure a high degree of anonymity and privacy.

3 Location Privacy

We will distinguish here between location privacy in location-based services and anonymization of trajectory data for their release.

3.1 Privacy in Location-Based Services

The massive use of mobile devices equipped with self-location technologies such as GPS has fostered the appearance of an unprecedented number of location-based services (LBS) that are gaining importance rapidly. The location-based applications that these new technologies can bring to people are almost unlimited and their advantages very substantial. However, the wide deployment of LBS can jeopardize the privacy of their users and raise social concern. Consequently, ensuring user privacy is essential to the success of those services.

We have mainly focused in TTP-free schemes and collaboration-based methods. [35, 36] refer to approximate location schemes. In [35] the authors proposed a method based on Gaussian noise addition to compute a fake location that is shared by k users. Thus, all k users use the same fake location and the LBS provider is unable to distinguish one user from the rest, so that their location becomes k-anonymous. This method was extended to support decentralized communications in [36].

On the other hand, [27] presented a new exact location method that has the advantages of these kind of methods such as pseudonymizers (i.e. simplicity and accuracy), and avoids their disadvantages (i.e. poor scalability and lack of privacy). The idea was to replace the classic concept of pseudonymizer, understood as a TTP, by a distributed pseudonymizer consisting of a set of collaborative users.

3.2 Trajectory Anonymization

Trajectories of mobile objects (individuals, cars, etc.) are routinely collected or at least collectible by such technologies as GPS, RFID, GSM, etc. The availability of trajectories, that is, mobility data, is extremely useful for public and corporate planning purposes. However, publication of original collected mobility data would result in obvious privacy disclosure: even if de-identified, trajectories are easily linkable to the individuals they correspond to and they tell a lot about that individual's lifestyle and habits. Furthermore, sensitive locations (hospitals, etc.) visited by individuals may be disclosed.

Domingo-Ferrer et al. [14] presented an anonymization method aimed at forming anonymized trajectories with true original locations and providing high utility properties but without a proven privacy level. In [16] the idea of trajectory anonymization by means of location permutation was leveraged, and two new methods were proposed that effectively satisfy provable privacy properties. Moreover, in [14] empirical results were obtained only on synthetic data, while in [16] experiments were added

that used a real-life data set of trajectories. Finally, in [40] the formalization of the notion of trajectory k-anonymity given in [16] was used to analyze the privacy offered by (k, δ)-anonymity; it was proven that (k, δ)-anonymity does not offer trajectory k-anonymity when $\delta > 0$, that is, when there is actual uncertainty. A direct implication of this result was that the methods that aimed at achieving (k, δ)-anonymity, *Never Walk Alone* (NWA, [1]) and *Wait for Me* (W4M, [2]) can offer trajectory k-anonymity only when $\delta = 0$ (when there is no uncertainty).

4 Differential Privacy

Differential privacy [12, 13] is a statistical disclosure control methodology based on output perturbation.The disclosure risk limitation offered by differential privacy is based on the limitation of the effect that any single individual has on a query response. If the influence of any single individual on the query response is small, publishing that response involves only a small disclosure risk for any individual. The problem with differential privacy is that achieving it normally results in very damaged data utility. Therefore, the research on differential privacy in ARES set out to find way to satisfy differential privacy that are more utility-preserving than those in the literature.

Any mechanism used to achieve differential privacy may be seen as the application of a perturbation to the real value of the query response. Soria-Comas and Domingo-Ferrer [30] introduced a mechanism to achieve differential privacy that worked by refining the prior knowledge/beliefs of the database user as much as possible, given the constraints set by differential privacy. This mechanism does not require complex computations and it guarantees that the response provides increased utility over the prior knowledge that the user had.

The original proposal [12, 13] to attain differential privacy masked the query response by adding a Laplace distributed noise whose magnitude is proportional to the global sensitivity of the query function. We showed in [31] that the Laplace distribution is not optimal, that is, that differential privacy can be reached with a noise distribution having a lower variance. In that paper, we built the optimal data-independent noise distribution with the help of an optimality criterion based on the concentration of the probability mass of the noise distribution around zero and we compared the resulting distribution with Laplace. For univariate query functions, both introduce a similar level of distortion; however, for multivariate query functions, optimal data-independent noise offers responses with substantially better data quality.

Other ARES contributions to the differential privacy literature highlight synergies between k-anonymity and differential privacy. Soria-Comas et al. [32] shows that the amount of noise required to fulfill differential privacy can be reduced if noise is added to a k-anonymous version of the data set, where k-anonymity is reached through a specially designed microaggregation of all attributes. On the other hand, [33] points out that, for data set anonymization, the t-closeness extension of k-anonymity is closely related to differential privacy.

5 Anti-discrimination in Data Mining

Along with privacy, discrimination avoidance is a very important issue when considering the legal and ethical aspects of data mining. It is more than obvious that most people do not want to be discriminated because of their gender, religion, nationality, age and so on, especially when those attributes are used for making automated decisions about them like giving them a job, loan, insurance, etc. Discovering such potential biases and eliminating them from the data used to train data mining classifiers without harming their decision-making utility is therefore highly desirable. For this reason, anti-discrimination techniques including discrimination discovery and prevention have been introduced in data mining.

Discrimination can be either direct or indirect. Direct discrimination occurs when decisions are made based on sensitive attributes. Indirect discrimination occurs when decisions are made based on nonsensitive attributes which are strongly correlated with biased sensitive ones.

In a first work [20], we introduced the initial idea of using rule protection and rule generalization for direct discrimination prevention, but we gave no experimental results. In [21], we introduced the use of rule protection in a different way for indirect discrimination prevention and we gave some preliminary experimental results. Finally, [22] presented a unified approach to direct and indirect discrimination prevention, with finalized algorithms and all possible data transformation methods based on rule protection and/or rule generalization that could be applied for direct or indirect discrimination prevention.

6 Conclusions

The overview that we have presented in this chapter is intended as a reading guide to the some of the contributions of ARES to data privacy technologies related to query profile protection, privacy in location-based systems, trajectory anonymization, differential privacy and anti-discrimination. Further details can be obtained by looking at the corresponding publications or at the other chapters in this book.

Acknowledgments The second author is partially supported by the Government of Catalonia through an ICREA Acadèmia Prize. The following partial supports are also gratefully acknowledged: the Spanish Government under projects CONSOLIDER INGENIO 2010 CSD2007-00004 “ARES” and TIN2011-27076-C03-01 “CO-PRIVACY”, and the European Commission under FP7 projects “DwB” and “Inter-Trust”. The second author is with the UNESCO Chair in Data Privacy, but the views expressed in this chapter neither necessarily reflect the position of UNESCO nor commit that organization.

References

1. Abul, O., Bonchi, F., Nanni, M.: Never walk alone: uncertainty for anonymity in moving objects databases. In: 24th International Conference on Data Engineering, pp. 376–385 (2008)
2. Abul, O., Bonchi, F., Nanni, M.: Anonymization of moving objects databases by clustering and perturbation. Inf. Syst. **35**(8), 884–910 (2010)
3. Bras-Amorós, M., Domingo-Ferrer, J., Stokes, K.: Configuraciones combinatóricas y recuperación privada de información por pares. In: Nuevos Avances en Criptografía y Codificación de la Información (2009)
4. Bras-Amorós, M., Stokes, K.: The semigroup of combinatorial configurations. Semigroup Forum **84**(1), 91–96 (2011)
5. Chor, B., Goldreich, O., Kushilevitz, E., Sudan, M.: Private information retrieval. In: IEEE Symposium on Foundations of Computer Science, pp. 41–50 (1995)
6. Chor, B., Gilboa, N., Naor, M.: Private Information Retrieval by keywords. Technical Report TR CS0917. Departament of computer Science, Technion (1997)
7. Chor, B., Goldreich, O., Kushilevitz, E., Sudan, M.: Private information retrieval. J. ACM **45**, 965–981 (1998)
8. Castellà, D., Romero-Tris, C., Viejo, A., Castellà-Roca, J., Solsona, F., Giné, F.: Diseño de una red P2P optimizada para la privatización de consultas en WSEs. In: XII Reunión Española sobre Criptología y Seguridad de la Información, pp. 273–278 (2012)
9. Castellà-Roca, J., Viejo, A., Herrera-Joancomartí, J.: Preserving users' privacy in web search engines. Comput. Commun. **32**(13), 1541–1551 (2009)
10. Domingo-Ferrer, J., Bras-Amorós, M.: Peer-to-peer private information retrieval. In: PSD 2008. LNCS, vol. 5262, pp. 315–323 (2008)
11. Domingo-Ferrer, J., Bras-Amorós, M., Wu, Q., Manjón, J.: User-private information retrieval based on a Peer-to-Peer community. Data Knowl. Eng. **68**(11), 1237–1252 (2009)
12. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Proceedings of the 3rd Theory of Cryptography Conference. LNCS, vol. 3876, pp. 265–284. Springer, New York (2006)
13. Dwork, C.: Differential privacy. In: Automata, Languages and Programming. LNCS, vol. 4052, pp. 1–12. Springer, New York (2006)
14. Domingo-Ferrer, J., Sramka, M., Trujillo, R.: Privacy preserving Publication of Trajectories using microaggregation. In: 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS (2010)
15. Domingo-Ferrer, J., Solanas, A., Castellà-Roca, J.: $h(k)$-private information retrieval from privacy-uncooperative queryable databases. Online Inf. Rev. **33**(4), 720–744 (2009)
16. Domingo-Ferrer, J., Trujillo-Rasua, R.: Microaggregation- and permutation-based anonymization of movement data. Inf. Sci. **208**, 55–80 (2012)
17. Erola, A., Castellà-Roca, J., Navarro-Arribas, G., Torra, V.: Semantic microaggregation for the anonymization of query logs. In: PSD 2010. LNCS, vol. 6344, pp. 127–137 (2010)
18. Erola, A., Castellà-Roca, J., Navarro-Arribas, G., Torra, V.: Semantic microaggregation for the anonymization of query logs using the open directory project. SORT-Statistics and Operations Research Transactions, pp. 41–58, Special issue (2011)
19. Erola, A., Castellà-Roca, J., Viejo, A., Mateo-Sanz, J.M.: Exploting social networks to provide privacy in personalized web search. J. Syst. Soft. **84**(10), 1734–1745 (2011)
20. Hajian, S., Domingo-Ferrer, J., Martínez-Ballesté, A.: Discrimination prevention in data mining for intrusion and crime detection. In: IEEE Symposium Series in Computational Intelligence in Cyber Security (2011)
21. Hajian, S., Domingo-Ferrer, J., Martínez-Ballesté, A.: Rule protection for indirect discrimination prevention in data mining. In: MDAI 2011. LNCS, vol. 6820, pp. 211–222 (2011)
22. Hajian, S., Domingo-Ferrer, J.: A methodology for direct and indirect discrimination prevention in data mining. IEEE Trans. Knowl. Data Eng. **25**(7), 1445–1459 (2013)

23. Howe, D.C., Nissenbaum, H.: TrackMeNot: resisting surveillance in web search. In: Kerr, I., Lucock, C., Steeves, V. (eds.) Lessons from the Identity Trail: Privacy, Anonymity and Identity in a Networked Society, pp. 409–428. Oxford University Press, Oxford UK (2009)
24. Jones, R., Kumar, R., Pang, B., Tomkins, A.: I know what you did last summer: query logs and user privacy. In: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, pp. 909–914 (2007)
25. Lee, J., Stinson, D.R.: A combinatorial approach to key predistribution for distributed sensor networks. In: Wireless Communications and Networking Conference-WCNC 2005, vol. 2, pp. 1200–1205 (2005)
26. Navarro-Arribas, G., Torra, V., Erola, A., Castellà-Roca, J.: User k-anonymity for privacy preserving data mining of query logs. Inf. Process. Manage. **48**(3), 476–487 (2012)
27. Pérez-Martínez, P.A., Solanas, A.: Location privacy through users' collaboration: a distributed pseudonymizer. In: Third International Conference on Mobile Ubiquitous Computing, Systems. Services and Technologies (2009)
28. Romero-Tris, C., Castellà-Roca, J., Viejo, A.: Multi-party private web search with untrusted partners. In: 7th International Conference on Security and Privacy in Communication Networks (2011)
29. Romero-Tris, C., Viejo, A., Castellà-Roca, J.: Improving query delay in private web search. In: International Workshop on Securing Information in Distributed Environments and Ubiquitous Systems (2011)
30. Soria-Comas, J., Domingo-Ferrer, J.: Sensitivity-independent differential privacy via prior knowledge refinement. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. **20**(6), 855–876 (2012)
31. Soria-Comas, J., Domingo-Ferrer, J.: Optimal data-independent noise for differential privacy. Inf. Sci. **250**, 200–214 (2013)
32. Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., Martínez, S.: Improving the utility of differentially private data releases via k-anonymity. In: 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (2013)
33. Soria-Comas, J., Domingo-Ferrer, J.: Differential privacy via t-closeness in data publishing. In: 11th Annual Conference on Privacy, Security and Trust, pp. 27–35 (2013)
34. Stinson, D.R.: Combinatorial designs: constructions and analysis. SIGACT News **39**(4), 17–21 (2008)
35. Solanas, A., Martínez-Ballesté, A.: Privacy protection in location-based services through a public-key privacy homomorphism. In: Euro PKI 2007. LNCS, vol. 4582, pp. 362–368 (2007)
36. Solanas, A., Martínez-Ballesté, A.: A TTP-free protocol for location privacy in location-based services. Comput. Commun. **31**(6), 1181–1191 (2008)
37. Stokes, K., Bras-Amorós, M.: Optimal configurations for Peer-to-Peer user-private information retrieval. Comput. Math. Appl. **59**(4), 1568–1577 (2010)
38. Stokes, K., Bras-Amorós, M.: Associating a numerical semigroup to the triangle-free configurations. Adv. Math. Commun. **5**(2), 351–371 (2011)
39. Stokes, K., Farràs, O.: Linear spaces and transversal designs: k-anonymous combinatorial configurations for anonymous database search. Des. Codes Crypt. **71**, 503–524 (2014)
40. Trujillo, R., Domingo-Ferrer, J.: On the privacy offered by k-d-anonymity. Inf. Syst. **38**(4), 491–494 (2013)
41. Viejo, A., Castellà-Roca, J.: Using social networks to distort users' profiles generated by web search engines. Comput. Netw. **54**(9), 1343–1357 (2010)
42. AOL Search Data Scandal. http://en.wikipedia.org/wiki/AOL_search_data_leak. Accessed Aug 2006
43. ODP. Open directory project. http://www.dmoz.org/
44. The Tor Project Inc: Tor: Overview. http://torproject.org/overview.html.en

Data Privacy: A Survey of Results

Vicenç Torra and Guillermo Navarro-Arribas

Abstract In this paper we present an overview of the results obtained by our research group within the area of data privacy. Results focus on data-driven problems (respondent and owner privacy with an unknown use) and user privacy. We have developed some new masking methods, developed methodologies for parameter selection, and developed some information loss and disclosure risk measures. We have also obtained important results on reidentification methods (record linkage) when used for disclosure risk assessment.

1 Introduction

Data privacy studies how to protect the privacy of individuals and corporations. The areas of Privacy Preserving Data Mining (PPDM) and Statistical Disclosure Control (SDC) study the theory, tools, and methodologies with this objective.

Data privacy methods and technologies can be classified according to different dimensions. One of them is on whose privacy is being sought (i.e., respondent, owner and user privacy). Another is on our knowledge of the type of computation a third party is interested to compute with the data (either we know what will be done or we do not know). See [1, 2] for a detailed description of the classification, and a survey on these areas. See also the following texts and monographies [3–7].

V. Torra (✉)
Institut d'Investigació en Intel·ligència Artificial,
Consejo Superior de Investigaciones Científicas Campus de la UAB,
08193 Bellaterra, Catalonia, Spain
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

G. Navarro-Arribas
Department of Information and Communications Engineering,
Universitat Autonoma de Barcelona, 08193 Bellaterra, Catalonia, Spain
e-mail: gnavarro@deic.uab.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_3

In this chapter we present the main results of our group in the field of data privacy organized according to the previous classification about whose privacy is being protected. As it will be seen below, most of our research has been focused towards data-driven methods for privacy protection. Those methods assume that we do not know the specific computation that will be performed on the data, and focus on methods to be applied by the data owner. This research is rooted on our initial research on the protection of numerical and categorical databases described in [8, 9]. We have developed methods and compared them with respect to information loss and disclosure risk.

In the last years we have obtained significant results on the analysis of risk using record linkage and other reidentification techniques. These results, that have been mainly applied to numerical databases, are equally applicable to other contexts.

In addition, although our work is mainly focused towards the protection of respondent privacy, we have also done some work on user privacy.

As some of our research is further explained in some of the chapters in this book. We will include links to these chapters in appropriate places.

The structure of the paper is as follows. In Sect. 2 we review the results on data driven (general-purpose) approaches. In Sect. 3 we describe our results on user-privacy. In Sect. 4 we briefly discuss the topic of knowledge-intensive data privacy. The chapter finishes with a summary.

2 Research on Data-Driven or General-Purpose Protection Methods

In the last years we have taken into account the three main issues related to data-driven methods:

- data protection methods, the methods that are needed to protect the data
- information loss measures, to evaluate in what extent a released data set can compromise privacy.
- disclosure risk.

Disclosure risk has been studied in the literature under different approximations. One approach considers risk a Boolean condition that, therefore, can only be satisfied or not. This is the case of k-anonymity and differential privacy. These definitions establish a level (established in terms of k in k-anonymity and ϵ in differential privacy) under which there is no problem with the risk. Another approach is to consider risk measurable and thus measures of risk are defined and studied. Definitions of risk based on uniqueness [10] and re-identification [11] correspond to the latter case. In this project we have studied disclosure risk measures based on re-identification.

I this section we review our research on masking methods, information loss and disclosure risk measures. We also show some last results on big data and large datasets.

2.1 Masking Methods and Data Types

We have considered the problem of privacy for different types of data. In particular, we have considered databases (either as simple numerical [12] or categorical [13] tables, or linked tables [14]), time series [15] and location privacy [16], logs (both search [17] and access [18, 19] logs), documents [20], and graphs and social networks [21–23]. We give details below for some of these data types.

2.1.1 Data Bases Files and Tables

We have considered the case of numerical and categorical data. In this area, following [9, 24] and others we have considered data privacy as a multicriteria optimization problem. The two competing criteria are information loss and disclosure risk. To solve this optimization problem we considered the use of genetic algorithms. These algorithms have been used to directly improve the database (*file optimization*) (see e.g. [12] and Chap. 6 [25] in this book) or to improve the parameters of a masking method (*parameter optimization*). The parameter optimization has been studied with the PRAM method. See [26–28] for references for PRAM and [13] for parameter optimization for PRAM.

Although the typical database studied in the literature is a single file with a set of *independent* variables, variables in databases are usually related (database schema establish relationships between variables), and databases have multiple linked tables. We have studied these two types of problems.

First, we considered the problem of files in which there are relationships between the variables. This is the case, for example, when a variable is a linear combination of others. Relationships between variables are known in the statistical community by edit constraints. Some data masking methods are not appropriate in this context because perturbation can cause violations to the constraints. This is the case of noise addition. We developed methods based on microaggregation [29] and noise addition [30] constraint-compliant. We also developed a system to automatically select an appropriate masking method when data was represented using XML and edit constraints were expressed using Schematron [31]. This type of problems were initially considered in [32].

Second, we have considered the problem of multiple releases from the same database. In [14] we reviewed the difficulties of publishing several copies of the same data protected using different approaches, and of several tables from the same database (linked tables). Algorithms were proposed for this purpose. This type of problem was previously considered by Nergiz et al. [33, 34].

The problem of multiple releases also appears in the case of dynamic data. That is, we have a database that changes with respect to time. In this case, when the data has to be published regularly, we need to take into account not only the possible disclosure of a single release but of the possible disclosure when all releases are combined. We considered this problem in [35] where the database consists of a set of documents, documents can be added or removed and a vector space of the documents in the database should maintain k-anonymity.

Besides of these problems we have also considered methods for typical databases. In particular, we have considered synthetic data generators based on fuzzy *c*-regression [36] and a variation of rank swapping for partial orders [37].

2.1.2 Time Series and Location Privacy

We studied the application of microaggregation to time series in [15, 38], considering and comparing the effect of using different distances between pairs of series when constructing cluster centers (e.g., Euclidean distance, Short time series distance). Reidentification algorithms [39] were used to compute disclosure risk. Information loss was evaluated in terms of some statistical analysis (e.g. ARMA) and methods for time series forecasting.

We also considered the problem of location privacy from a time series perspective, both when the sequences are defined in terms of (*numerical*) locations [16] as well as when sequences correspond to sequences of events (*categorical* locations). The case of sequences of events required the definition and selection of similarity measures [40] and aggregation functions [41, 42] for these sequences.

2.1.3 Search and Access Logs

We have tackled the problem of ensuring privacy in web usage mining by protecting access logs, which is the main source for such mining. In [18, 19] we used microaggregation to provide privacy-preserving data mining of typical access logs generated by a Web server.

Another interesting problem is that of ensuring privacy in search or query logs. That is, logs generated by a web search engine. A well known incident regarding query logs released by AOL with a poor anonymization [43] showed the need for proper protection techniques for these specific logs. In this line we have applied microaggregation at user level in search logs [17] and taking into account the semantics of the queries based on the open directory project [44].

2.1.4 Documents

We have considered the sanitization of documents. We have considered two types of problems. On the one hand, the case of privacy in indices [20] built from the documents. That is, we have a set of documents indexed by vectors of words and we want these vectors to be *k*-anonymous. To achieve this purpose, we used both syntactic and semantic distances (e.g., WordNet). This problem was reconsidered in [35] for dynamic sets of documents.

On the other hand, we studied the sanitization of documents [45, 46] (see also Chap. 9 in this book [47]). Detecting terms that may compromise privacy. In [45] we used an entity recognition system to identify the parts of the document to be protected.

2.1.5 Social Networks

We have investigates several different problems related to the privacy of social networks and of graphs, as their mathematical representation. But also about the protection of the information attached to the graphs as in [48].

In [23] we considered the problem of disclosure risk assessment and more specifically record linkage and k-anonymity for graphs. The problem of k-degree anonymity has been considered from the theoretical viewpoint [49] and from the computational viewpoint [21] (see also Chap. 13 in this book [50]). We have also explored [51] the relationship between graphs and the concept of p-stability. Once a privacy model is selected, we have explored tools as clustering [52], edge modification [53], as well as other perturbation techniques [22, 54] to achieve a protected version of the social network.

2.2 Information Loss Measures

We have defined information loss measures and used them for various types of data. In particular, we have considered measures based on clustering, classification and probability distributions. Clustering based measures have been used on different types of data. For example, in numerical [55] and categorical [56] data for files, logs [17], and graphs for social networks [53]. Classification based measures in [57] showed that masked data is still useful for building classifiers. Hellinger distance between probability distributions and entropy have been considered in relation to categorical data [58].

2.3 Disclosure Risk Measures

We have studied measures of disclosure when they are defined in terms of reidentification. That is, the measure is proportional to the number of links between intruders data and released data. On the one hand we have used this approach extensively under a variety of contexts to evaluate data protection measures. On the other we have obtained new results with respect to the algorithms for record linkage.

We have introduced a formalization of re-identification algorithms based on imprecise probabilities. Good masking methods should be resilient to attacks using all re-identification algorithms. In order to study this approach, we need to define mathematically what a re-identification algorithm is. In [59, 60] we proposed a definition based on imprecise probabilities and gave some examples of their application. The definitions imply that algorithms that are not consistent with the available knowledge are not considered as re-identification algorithms and, thus, are not a threat with respect to disclosure risk.

2.3.1 The Worst-Case Scenario

We have defined and studied disclosure risk in the worst case scenario. In this scenario an intruder knows the optimal parameters of a re-identification algorithm. In order to evaluate this scenario with actual files, we have estimated the optimal parameter using machine learning and optimization algorithms. We have studied this approach assuming different distances in a distance-based record linkage. In particular, weighted distances using a weighted mean [61, 62], an OWA operator [62], a Choquet integral [63] and a bilinear form have been used (see e.g. [64]).

2.3.2 Transparency and Disclosure Risk

There is transparency in data privacy when a data set is released with information on how this data set has been protected as well as with any information on the parameterizations used. Transparency implies that adversaries can use all this available information about masking methods and parameters to better attack the data. We have studied specific record linkage algorithms to measure better disclosure risk. Specific record linkage are tailored to a particular masking method. In particular, we have developed record linkage algorithms to attack data protected using rank swapping (rank swapping specific record linkage [65]) and microaggregation (microaggreation specific record linkage [66, 67]). Then, we have developed variations of rank swapping which are resilient to transparency attacks.

2.4 Big Data and Large Datasets

While most of our results deal with datasets of a regular size, we have also developed some methods for large and very large datasets. Methods for social networks and dynamic data described above fall in this area. In addition to this work, we have also developed methods for masking and measuring disclosure risk for standard but very large databases. See e.g. [68] for an overview of this problem, [69] for a microaggregation approach appropriate for large numerical data volumes, and [70] for a discussion and an algorithm for measuring disclosure of large datasets.

3 User Privacy

In user privacy the user has an active role to protect his own privacy. We have considered the problem of a user sending queries to a search engine trying to protect his privacy with respect to the owner of the search engine.

Our approach considered that what makes a person unique is the combination of his different interests. Each of the interests alone are common in other people. In order to implement this approach we developed a privacy agent as a browser extension that used different virtual identities, one for each interest. Then, given a query, the agent classifies it to an interest and its corresponding virtual identity. In

this way, the user can achieve a certain level of privacy (the level depends on e.g. the number of virtual identities) and at the same time some personalization (each identity has personalization on the queries submitted). The agent, called *Dissociating Privacy Agent* (DisPA for short) was presented in [71] (see also Chap. 21 [72] in this book).

4 Knowledge-Intensive Data Privacy

In [73] we plead for the development of knowledge-intensive tools for data privacy. We consider that masking methods need to take into consideration the semantics of the data being protected, the schema of the databases and the relationships between the variables. Similarly, we consider that risk measures have to take into account the semantics of the data, as well as any knowledge on the masking methods.

Some of the results described in previous sections fall into knowledge-intensive data privacy. This is the case, for example, of masking methods for logs and documents [44, 46, 74]. They need to use ontologies and dictionaries. Chapter 7 [75] in this book also focus on this topic.

The masking methods discussed above where the protection takes into account the constraints between variables can also be classified as knowledge-intensive data privacy.

5 Summary

In this chapter we have reviewed the main topics of research that were studied by our team during the ARES project. Table 1 provides an overview of the contributions described in this chapter.

Table 1 Summary of contributions

Data driven	Masking	DB files, tables	[12–14, 25, 29–31, 35–37]
		Time series / location	[15, 16, 38–42]
		Logs	[17–19, 44]
		Documents	[20, 35, 45–47]
		Social networks	[21–23, 48–54]
	Information loss		[17, 53, 55–58]
	Disclosure risk	Formalization	[59, 60]
		Worst-case	[61–64]
		Transparency	[65–67]
	Big data		[68–70]
User privacy			[71, 72]
Knowledge			[44, 46, 73–75]

Acknowledgments The research leading to these results was mainly funded by the Spanish MEC projects ARES (CONSOLIDER INGENIO 2010 CSD2007-00004). Partial support from Spanish projects e-Aegis (TSI2007-65406-C03), COPRIVACY (TIN2011-27076-C03-03), and from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n? 262608 is also acknowledged.

References

1. Navarro-Arribas, G., Torra, V.: Information fusion in data privacy: a survey. Inf. Fusion **13**(4), 235–244 (2012)
2. Torra, V., Navarro-Arribas, G.: Data Privacy, WIREs Data Mining and Knowledge Discovery, in press (2014)
3. Doyle, P., Lane, J.I., Theeuwes, J.J.M., Zayatz, L. (eds.): Confidentiality. Disclosure and Data Access, Theory and Practical Applications for Statistical Agencies, North-Holland (2001)
4. Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Nordholt, E.S., Spicer, K., de Wolf, P.-P.: Statistical Disclosure Control. Wiley, New York (2012)
5. Torra, V.: Data Privacy, Springer, Berlin. See also http://www.ppdm.cat/dp (2014)
6. Vaidya, J., Clifton, C.: Zhu, M.: Privacy Preserving Data Mining, Springer (2006)
7. Willenborg, L., de Waal, T.: Elements of Statistical Disclosure Control. Lecture Notes in Statistics, Springer, Berlin (2001)
8. Domingo-Ferrer, J., Torra, V.: Disclosure control methods and information loss for microdata. In: Doyle, P., Lane, J.I., Theeuwes, J.J.M., Zayatz, L. (eds.) Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 91–110. Elsevier Science (2001)
9. Domingo-Ferrer, J., Torra, V.: A quantitative comparison of disclosure control methods for microdata. In: Doyle, P., Lane, J.I., Theeuwes, J.J.M., Zayatz, L. (eds.) Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 111–134. North-Holland (2001)
10. Elliot, M.J., Skinner, C.J., Dale, A.: Special uniqueness. random uniques and sticky populations: some counterintuitive effects of geographical detail on disclosure risk. Res. Official Stat. **1**(2), 53–67 (1998)
11. Winkler, W.E.: Re-identification methods for masked microdata, PSD 2004. Lect. Notes Comput. Sci. **3050**, 216–230 (2004)
12. Jimenez, J., Marés, J., Torra, V.: An evolutionary approach to enhance data privacy. Soft Comput. **15**(7), 1301–1311 (2011)
13. Marés, J., Torra, V.: An Evolutionary Algorithm to Enhance Multivariate Post-Randomization Method (PRAM) Protections, Information Sciences, in press (2014)
14. Stokes, K., Torra, V.: Multiple releases of *k*-anonymous data sets and *k*-anonymous relational databases. Int. J. Unc. Fuzziness Knowl. Based Syst. **20**(6), 839–853 (2012)
15. Nin, J., Torra, V.: Towards the evaluation of time series protection methods. Inf. Sci. **179**(11), 1663–1677 (2009)
16. Martínez-Bea, S., Torra, V.: Trajectory anonymization from a time series perspective. In: Proceedings IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011), pp. 401–408 (2011)
17. Navarro-Arribas, G., Torra, V., Erola, A., Castellà-Roca, J.: User k-anonymity for privacy preserving data mining of query logs. Inf. Process. Manage. **48**(3), 476–487 (2012)
18. Navarro-Arribas, G., Torra, V.: Tree-based Microaggregation for the Anonymization of Search Logs. In: 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (Workshop on Soft approaches to information access on the Web), vol. 3, Milan, Italy, IEEE, pp. 155–158 (2009)

19. Navarro-Arribas, G., Torra, V.: Privacy-preserving data-mining through microaggregation for web-based e-commerce. Internet Res. **20**(3), 366–384 (2010)
20. Abril, D., Navarro-Arribas, G., Torra, V.: Vector space model anonymization. In: Proceedings of CCIA (2013)
21. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: An algorithm for *k*-degree anonymity on large networks. In: Proceedings of 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (2013)
22. Nettleton, D. F., Torra, V., Dries, A.: The effect of constraints on information loss and risk for clustering and modification based graph anonymization methods, arXiv preprint arXiv:1401.0458 (2014)
23. Stokes, K., Torra, V.: Reidentification and k-anonymity: a model for disclosure risk in graphs. Soft Comput. **16**(10), 1657–1670 (2012)
24. Sebé, F., Domingo-Ferrer, J., Mateo-Sanz, J.M., Torra, V.: Post-masking optimization of the tradeoff between information loss and disclosure risk in masked microdata sets. Lect. Notes Comput. Sci. **2316**, 187–196 (2002)
25. Marés, J., Torra, V., Shlomo, N.: Optimisation-Based Study of Data Privacy by Using PRAM. Chapter 6, Advanced Research on Data Privacy. Springer, Berlin (2014)
26. De Wolf, P.P., Van Gelder, I.: An empirical evaluation of PRAM. Discussion paper 04012. Statistics Netherlands, Voorburg/Heerlen (2004)
27. Gouweleeuw, J.M., Kooiman, P., Willenborg, L.C.R.J., De Wolf, P.-P.: Post Randomisation for Statistical Disclosure Control: Theory and Implementation', Journal of Official Statistics, vol. 14, pp. 4 463–478. Also as Research Paper No. 9731. Statistics Netherlands, Voorburg (1997)
28. Gross, B., Guiblin, P., Merrett, K.: Implementing the Post Randomisation method to the individual sample of anonymised records (SAR) from the 2001 Census, paper presented at "The Samples of Anonymised Records, An Open Meeting on the Samples of Anonymised Records from the 2001 Census". http://www.ccsr.ac.uk/sars/events/2004-09-30/gross.pdf (2004)
29. Torra, V.: Constrained microaggregation: adding constraints for data editing. Trans. Data Priv. **1**(2), 86–104 (2008)
30. Cano, I., Torra, V.: Edit constraints on microaggregation and additive noise. Lect. Notes Comput. Sci. **6549**, 1–14 (2011)
31. Cano, I., Navarro-Arribas, G., Torra, V.: A new framework to automate constrained microaggregation. In: Proceedings PAVLAD Workshop in CIKM, pp. 1–8 (2009)
32. Shlomo, N., De Waal, T.: Protection of micro-data subjecto to edit constraints against statistical disclousure. J. Official Stat. **24**(2), 229–253 (2008)
33. Nergiz, M.E., Clifton, C., Nergiz, A.E.: MultiRelational k-Anonymity. Proc. ICDE **2007**, 1417–1421 (2007)
34. Nergiz, M.E., Clifton, C., Nergiz, A.E.: MultiRelational k-anonymity. IEEE Trans. Knowl. Data Eng. **21**, 1104–1117 (2009)
35. Navarro-Arribas, G., Abril, D., Torra, V.: Dynamic anonymous index for confidential data. In: Proceedings DPM 2013. Lecture Notes in Computer Science, vol. 8247, pp. 362–368 (2014)
36. Cano, I., Torra, V.: Generation of synthetic data by means of fuzzy c-regression. In: Proceedings of FUZZ-IEEE, pp. 1145–1150 (2009)
37. Torra, V.: Rank swapping for partial orders and continuous variables. In: Proceedings ARES 2009, WAIS Workshop, pp. 888–893 (2009)
38. Nin, J., Torra, V.: Extending microaggregation procedures for time series protection. Lect. Notes Comput. Sci. **4259**, 899–908 (2006)
39. Nin, J., Torra, V.: Distance based re-identification for time series. Analysis of distances. Lect. Notes Comput. Sci. **4302**, 205–216 (2006)
40. Gómez-Alonso, C., Valls, A.: A similarity measure for sequences of categorical data based on the ordering of common elements. LNAI **5285**, 134–145 (2008)
41. Valls, A., Gómez-Alonso, C., Torra, V.: Generation of prototypes for masking sequences of events. In: Proceedings ARES 2009, WAIS Workshop, pp. 947–952 (2009)
42. Valls, A., Nin, J., Torra, V.: On the use of aggregation operators for location privacy. In: Proceedings IFSA-EUSFLAT, pp. 489–494 (2009)

43. Barbaro, M., Zeller, T.: A Face Is Exposed for AOL Searcher No. 4417749, The New York Times, August 9, 2006. Retrieved April 25, 2010 (2006)
44. Erola, A., Castellà-Roca, J., Navarro-Arribas, G., Torra, V., (2011) Semantic microaggregation for the anonymization of query logs using the open directory project. SORT - Statistics and Operations Research Transactions, pp. 41–58.
45. Abril, D., Navarro-Arribas, G., Torra, V.: On the declassification of confidential documents. Lect. Notes Comput. Sci. **6820**, 235–246 (2011)
46. Nettleton, D., Abril, D.: Document sanitization: Measuring search engine information loss and risk of disclosure for the wikileaks cables, LNCS 7556 (2012)
47. Nettleton, D.F., Abril, D.: An Information Retrieval Approach to Document Sanitization. Chapter 9, Advanced Research on Data Privacy. Springer, Berlin (2014)
48. Marés, J., Torra, V.: On the protection of social networks user's information. Knowl.-Based Syst. **49**, 134–144 (2013)
49. Salas, J., Torra, V.: Approximating degree sequences with regular graphic sequences, manuscript (2014)
50. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: A Summary of k-Degree Anonymous Methods for Privacy-Preserving on Networks. Chapter 13, Advanced Research on Data Privacy. Springer, Berlin (2014)
51. Torra, V., Shafie, T.: Data protection for online social networks and p-stability for graphs, manuscript (2014)
52. Stokes, K., Torra, V.: On some clustering approaches for graphs. In: Proceedings IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011), pp. 409–415 (2011)
53. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: Analyzing the impact of edge modications on networks. Lect. Notes Comput. Sci. **8234**, 296–307 (2013)
54. Nettleton, D.F., Torra, V., Dries, A.: A comparison of clustering and modification based graph anonymization methods with constraints. Int. J. Comput. Appl. (2014)
55. Cano, I., Ladra, S., Torra, V.: Evaluation of information loss for privacy preserving data mining through comparison of fuzzy partitions. In: Proceedings of FUZZ-IEEE 2010/WCCI (2010)
56. Marés, J., Torra, V.: Clustering-based categorical data protection. Lect. Notes Comput. Sci. **7556**, 78–89 (2012)
57. Herranz, J., Matwin, S., Nin, J., Torra, V.: Classifying data from protected statistical datasets. Comput. Secur. **29**(8), 875–890 (2010)
58. Torra, V., Carlson, M.: On the Hellinger distance for measuring information loss in microdata, UNECE / Eurostat Work Session on Statistical Confidentiality, 8th Work Session 2013. Ottawa, Canada (2013)
59. Torra, V., Stokes, K.: A formalization of re-identification in terms of compatible probabilities, arXiv preprint arXiv:1301.5022 (2013)
60. Torra, V., Stokes, K.: A formalization of record linkage and its application to data protection. Int. J. Unc. Fuzziness Knowl. Based Syst. **20**(6), 907–919 (2012)
61. Abril, D., Navarro-Arribas, G., Torra, V.: Improving record linkage with supervised learning for disclosure risk assessment. Inf. Fusion **13**(4), 274–284 (2012)
62. Torra, V., Navarro-Arribas, G., Abril, D.: Supervised learning for record linkage through weighted means and OWA operators. Control Cybern. **39**(4), 1011–1026 (2010)
63. Abril, D., Navarro-Arribas, G., Torra, V.: Choquet integral for record linkage. Ann. Oper. Res. **195**, 97–110 (2012)
64. Abril, D., Torra, V., Navarro-Arribas, G.: Supervised Learning Using a Symmetric Bilinear Form for Record Linkage, manuscript (2014)
65. Nin, J., Herranz, J., Torra, V.: Rethinking rank swapping to decrease disclosure risk. Data Knowl. Eng. **64**(1), 346–364 (2008)
66. Nin, J., Herranz, J., Torra, V.: On the disclosure risk of multivariate microaggregation. Data Knowl. Eng. **67**, 399–412 (2008)
67. Nin, J., Torra, V.: Analysis of the univariate microaggregation disclosure risk. New Gener. Comput. **27**, 177–194 (2009)

68. Muntés-Mulero, V., Nin, J.: Privacy and anonymization for very large datasets. In: Proceedings 18th ACM conference on CIKM (2009)
69. Solé, M., Muntés-Mulero, V., Nin, J.: Efficient microaggregation techniques for large numerical data volumes. Int. J. Inf. Secur. **11**(4), 253–267 (2012)
70. Herranz, J., Nin, J., Solé, M.: Kd-trees and the real disclosure risks of large statistical databases. Inf. Fusion **13**, 260–273 (2012)
71. Juárez, M., Torra, V.: Toward a privacy agent for information retrieval. Int. J. Intel. Syst. **28**(6), 606–622 (2013)
72. Juárez, M., Torra, V.: Optimisation-Based Study of Data Privacy by Using PRAM. Chapter 21, Advanced Research on Data Privacy. Springer, Berlin (2014)
73. Torra, V.: Towards knowledge intensive data privacy. Data privacy management and autonomous spontaneous security. Lect. Notes Comput. Sci. **6514**, 1–7 (2011)
74. Abril, D., Navarro-Arribas, G., Torra, V.: Towards semantic microaggregation of categorical data for confidential documents. Lect. Notes Comput. Sci. **6408**, 266–276 (2010)
75. Martínez, S., Valls, A., Sanchez, D. Semantic anonymisation of categorical datasets. Chapter 7, Advanced Research on Data Privacy. Springer, Berlin (2014)

Part II
Respondent Privacy: SDC and PPDM

A Review of Attribute Disclosure Control

Stan Matwin, Jordi Nin, Morvarid Sehatkar and Tomasz Szapiro

Abstract Attribute disclosure occurs when the adversary can infer some sensitive information about an individual without identifying individual's record in the published data set. To address this issue several privacy models were proposed with the goal of increasing the uncertainty of the adversary in deriving sensitive information from published data. In this chapter, firstly we review the underlying scenario used in statistical disclosure control (SDC) and Privacy-Preserving Data Mining (PPDM). In this chapter, we describe the attribute disclosure underlying scenario, the different forms of background knowledge of the adversary the adversary may have and their potential privacy attacks. then, we review the approaches introduced in the literature to tackle attribute disclosure attacks.

1 The Underlying Scenario

We now proceed with a more formal description of the scenario that guides our presentation.

S. Matwin
Faculty of Computer Science, Dalhousie University, Halifax, Canada
e-mail: stan@dal.ca

S. Matwin
Institute for Computer Science, Polish Academy of Sciences, Warsaw, Poland

J. Nin (✉)
Barcelona Supercomputing Center (BSC), Universitat Politècnica de Catalunya (BarcelonaTech), Barcelona, Catalonia, Spain
e-mail: nin@ac.upc.edu

M. Sehatkar
School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Ontario, Canada
e-mail: msehatkar@uottawa.ca

T. Szapiro
Division of Decision Analysis and Support, Warsaw School of Economics, Warsaw, Poland
e-mail: tszapiro@gmail.com

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_4

A data set X can be seen as a matrix with n rows (*records*) and k columns (*attributes*). Each row contains the values of the attributes for an individual. The attributes in a data set can be classified in three categories:

- **Identifiers** They are attributes which unambiguously identify the individual, for example, the passport number.
- **Quasi-identifiers** They are attributes which can identify the individual when some of those attributes are combined. For example, city of born, city or job cannot identify an individual, but the set of individuals working at the Daily Planet, living in Metropolis and being born at Smallville, contains a single individual, superman.
- **Confidential attributes** They are attributes which contain sensitive information about the individual. For example, salary.

When considering this classification, a data set X is defined as $X = id||X_{nc}||X_c$, where id are the identifiers, X_{nc} are the non-confidential quasi-identifier attributes, and X_c are the confidential attributes. Normally, before releasing a data set X with confidential attributes, a protection method ρ is applied, leading to a protected data set X'. Indeed, we will assume the following typical scenario: (i) identifier attributes in X are either removed or encrypted, therefore we will write $X = X_{nc}||X_c$; (ii) confidential attributes X_c are not modified, and so we have $X'_c = X_c$; (iii) the protection method itself is applied to non-confidential quasi-identifier attributes, in order to preserve the privacy of the individuals whose confidential data is being released. Therefore, we have $X'_{nc} = \rho(X_{nc})$. This scenario allows third parties to have precise information on confidential data without revealing to whom the confidential data belongs to.

Since non-confidential quasi-identifier values are modified when a privacy protection method is applied on data set, valuable knowledge might be lost to favor data owners privacy. A good protection method must achieve a good trade-off between data utility and privacy. In other words, the protected data set X' must be:

- Close enough to X such that the knowledge and statistics extracted on X' will be very similar to those that would be obtained by computing directly on X. In other words, the (statistical) *information loss* that appears in the transition from X to X' must be small.
- Different enough from X such that an attacker has a (very) small probability to obtain any correct relation between a protected record in X' with the quasi-identifier attributes corresponding to this record. This probability is denoted as the *disclosure risk*.

Information loss (IL) measures the statistical utility of the protected data set X', comparing its usefulness with respect to the one of the original data set X. A few different approaches are used to calculate the information loss. In [1] the authors calculate the average divergence of some statistical values when they are computed on both the original and the protected data sets. A probabilistic variation of these measures (PIL) was presented in [2] to ensure that the information loss value is always within the interval [0, 1]. When a privacy protection method does partial suppressions

or detail reductions (generalizations) on the original data, different measures as the Global Certainty Penalty (GCP) measure [3] are considered.

2 Background Knowledge of the Adversary

The simplest form of background knowledge an adversary may pose is the values of quasi-identifiers of an individual. Employing this knowledge, the adversary may be able to directly infer an individual's sensitive value(s) based on the distribution of values of sensitive attribute(s) in the equivalence class where individual's record belongs to. Besides quasi-identifiers an adversary may have more complex forms of background knowledge that enables her for attribute disclosure attacks.

Machanavajjhala et al. [4, 5] considered the background knowledge that can be modeled by *negation statements* [6]. For instance, "men do not have cervical cancer", or "Bob never travels, thus he is extremely unlikely to have Ebola", or "Japanese have a very low incidence of heart disease" [5].

Another form of adversary's background knowledge, which can not be expressed by negation statements, was illustrated by Martin et al. [6] through the following example.

Example [6] Assume that a married couple, Charlie and Hannah, are neighbors of Alice and she saw that they both were taken to the hospital. Alice knows that Hannah had a flu shot recently but Charlie did not. Therefore she believes that Charlie's immunity to the flu is less than Hannah. Since Hannah and Charlie are living together, Alice can infer that if Hannah could not resist against flu, then it is most probably that Charlie could not as well. So Alice can infer that Charlie has flu.

Motivated by this, Martin et al. [6] introduced a *language* to represent background knowledge of the adversary. Instead of representing the specific content of adversary's knowledge, this language quantifies the *amount* of knowledge of the adversary [7]. It is defined as the set of all possible conjunctions of k implications [7]. For example, the knowledge that "if Hannah has the flu, then Charlie also has the flu", will be represented as

$$t_{Hannah}[Disease] = flu \quad \rightarrow \quad t_{Charlie}[Disease] = flu$$

where $t_X[S]$ is value of sensitive attribute S for individual X in a record t.

The adversary's background knowledge which is in the form of negation statement can also be represented by this language. For example, the knowledge that "Bob does not have Ebola" can be represented as

$$t_{Bob}[Disease] = Ebola \quad \rightarrow \quad t_{Bob}[Disease] = OvarianCancer$$

Although the knowledge representation language proposed by Martin et al, provides a general purpose framework to capture knowledge of the adversary, it has

several shortcomings that are brought up by Chen et al. [8]. The most important problem discussed in [8] is that the data publisher can not easily understand the practical meaning of k implications and, therefore, quantifying the knowledge is not intuitive. Chen et al also show that some important types of knowledge can not practically be expressed by this language. Having this argument, Chen et al proposed the language $L_{t,s}(l, k, m)$ to express the knowledge of the adversary in a more intuitive way. In this new language, knowledge of the adversary is represented by three factors l, k, and m based on three facts the adversary knows about an individual t:

- l indicates that the adversary has l pieces of knowledge about individual t
- k indicates that the adversary has information about k individuals other than t
- m indicates that the adversary knows m individuals such that if any of them has sensitive value s then t has value s, for instance if s is a contagious disease

Another form of background knowledge of the adversary leading to attribute disclosure is proposed by Li et al. [9]. The authors mine negative association rules from the data as the background knowledge of the adversary and then anonymize the data in such a way that eliminates this knowledge from the data. The idea is that the background knowledge of the adversary reflects itself in the data, therefore, data mining approaches should be able to extract this knowledge. Although in this technique only the negative association rules are considered as the background knowledge of the adversary, Li et al talk about the possibility of discovering the other types of background knowledge from the data as long as it does not have any negative effect on the utility of data [9].

Wong et al. [10] considered adversaries with the knowledge of the mechanism or anonymization algorithm employed for publishing data. They recognized that based on such knowledge and the fact that the main goal of all anonymization techniques is to reduce information loss, the adversary may be able to infer sensitive information.

Having *probabilistic knowledge* about one part of the domain can also empower the adversary for attribute disclosure attacks [7]. For example the adversary may know the distribution of values of a sensitive attribute in a (or part of a) population, such as "the rate of cancer in Gotham City is only 10 %, but it is higher(about 50 %) if only males in Gotham City are considered" [7].

In the next section, we talk about the potential privacy attacks based on different background knowledge of the adversaries presented in this section.

3 Privacy Attacks

Different types of privacy attacks, leading to disclosure of sensitive value(s) of individuals, are introduced in the literature. These attacks will be discussed in this section.

Homogeneity attack [4, 5] When all or majority of the records in an equivalence class [1] in an anonymized data set have an identical value for a sensitive attribute the data set is vulnerable to a homogeneity attack.

In this attack, the adversary uses her background knowledge about the quasi identifiers of an individual to find the equivalence class where the individual's record belongs. Then if all or most of the sensitive values in this equivalence class are the same, e.g s_m, without needing to re-identify the record of that individual the adversary will be able to infer, with a high confidence, that the value of that sensitive attribute for that individual is s_m.

This attack is illustrated by Machanavajjhala et al. [4, 5] through the following example:

Example (homogeneity attack): Alice and Bob are neighbors and one day Alice sees that Bob is taken to the hospital by ambulance. While Bob is in the hospital, Alice discovers Table 1b, an anonymized version version of the data set in Table 1, containing current inpatient records published by the hospital. Therefore, she knows that one of the records in the Table belongs to Bob and she tries to figure out what Bob's disease is. As she is Bob's neighbor, she knows that Bob is American and he is 31 years old. She also knows that he is living in the zip code 13053 (the same as herself). Therefore, she can infer that one of the records in the last equivalence class, i.e. records 9, 10, 11, or 12, belongs to Bob. As all patients in that equivalence class have cancer, Alice, without any extra effort to re-identify Bob's record, will infer that Bob has cancer and therefore she will jeopardize his privacy.

The above example shows that when there is lack of diversity in the values of a sensitive attribute in an equivalence class the privacy of the individuals can be violated.

Background knowledge attack [4, 5] Adversaries can launch this attack if, besides quasi-identifiers, they pose some extra background knowledge about an individual. Machanavajjhala et al. [4, 5] consider adversaries whose knowledge is in the form of negation statements. With such knowledge, an adversary is able to eliminate some (in special case, all except one) of the sensitive values in an equivalence class and therefore increase her certainty about the value of a sensitive attribute in the record of an individual belonging to that equivalence class. The authors demonstrated this attack with the following example in which the adversary was able to eliminate all sensitive values in an equivalence class except one value, using her background knowledge:

Example (background knowledge attack) [4, 5]: Alice has a pen-friend named Umeko who is a 21 years old Japanese girl living in the zip code 13068. Alice knows that Umeko is admitted to the same hospital as Bob and, therefore, her record is in the published anonymized data in Table 1b. By looking at the data, Alice will know that Umeko's record resides in the first equivalence class and one of the records

[1] We define an equivalence class of an anonymized table to be a set of records that have the same values for the non-confidential quasi-identifiers.

Table 1 Inpatient data [25]

(*a*) *Original data* [5]

	Non-sensitive			Sensitive
	Zip code	Age	Nationality	Condition
1	13053	28	Russian	Heart disease
2	13068	29	American	Heart disease
3	13068	21	Japanese	Viral infection
4	13053	23	American	Viral infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart disease
7	14850	47	American	Viral infection
8	14850	49	American	Viral infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

(*b*) 4-*anonymous data* [5]

	Non-sensitive			Sensitive
	Zip code	Age	Nationality	Condition
1	130**	<30	*	Heart disease
2	130**	<30	*	Heart disease
3	130**	<30	*	Viral infection
4	130**	<30	*	Viral infection
5	1485*	≥40	*	Cancer
6	1485*	≥40	*	Heart disease
7	1485*	≥40	*	Viral infection
8	1485*	≥40	*	Viral infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

1, 2, 3, or 4 belongs to Umeko. Without any extra information, Alice can not conclude if Umeko has viral infection or heart disease. However, it is well-known that heart disease is very rare among Japanese because of their diet. This additional information enables Alice to infer that it is highly probable that Umeko is in the hospital because of a viral infection.

As we discussed in Sect. 2, there are more complex forms of background knowledge than the knowledge which can be modeled by negation statements. Consequently, several forms of background knowledge attacks can be launched by the adversaries. To deal with each of such attacks a particular solution is proposed in the literature that we will discuss in the next section.

Skewness attack [11] This attack occurs when the overall distribution of values of a sensitive attribute is skewed. The following example illustrates this attack:

Example (skewness attack) [11]: consider a data set containing records of 10,000 patients with a set of quasi identifiers and one sensitive attribute for test result of a virus with two possible values "Positive" and "Negative". Also, assume that test results for 99 % of those patients are "Negative" and for just 1 % are "Positive". An equivalence class that has equal number of positive and negative cases will be a violation to privacy since each patient belonging to this class will be considered as a positive case with probability of 50 % while this probability in the whole data set is just 1 %.

Similarity attack [11] When the values of sensitive attribute in an equivalence class are different but semantically similar, the adversaries can attack the privacy of individuals. This attack is called a similarity attack and is shown with the following example:

Example (Similarity attack) [11] consider a data set with one sensitive value Disease. Assume there is an equivalence class containing three records and the values of attribute Disease for these three records are gastric ulcer, gastritis, and stomach cancer. If Alice knows that Bob's record is in this equivalence class, then, without needing to re-identify Bob's record, she can infer that Bob has a stomach-related disease since all values of attribute Disease in this equivalence class are stomach related. This inference might be a breach of privacy for Bob. This can be also a problem in the case of numerical sensitive attributes. For instance, consider the data set has also a sensitive attribute Salary and in one of the equivalence classes, containing three records, the values of attribute Salary in those three records are 3, 4, and 5K. If Alice knows that Bob's record is in this equivalence class, she can infer that Bob's salary is low since all values of attribute Salary in this equivalence class are in the range [3–5K].

Proximity breach [12] Generally speaking, a privacy breach on a numerical sensitive attribute occurs even if an adversary can infer a *close* value to the exact value of the attribute. This is opposed to privacy violations on categorical attributes that the adversary needs to infer the exact value of sensitive attributes. Besides [11], there are some other works aiming to capture the semantic knowledge of an adversary about a numeric attribute [13, 14]. Li et al. [12] summarized all such privacy breaches on numeric attributes and presented a privacy attack, called *proximity breach*.

Minimality attack [10] Another type of privacy attack, proposed by Wong et al. [10], is *minimality attack* which is possible if the adversary has knowledge about the mechanism or algorithm of data anonymization. This attack is based on an implicit principle that all anonymization techniques follow. According to this principle the amount of data distortion in any anonymization process must be always minimum [15] and data modification, using generalization, suppression, etc, should not be done more than necessary.

Table 2 Minimality attack [10]

(*a*) *Original data*	
QID	Disease
q1	HIV
q1	HIV
q2	Non-sensitive
q2	Non-sensitive
q2	Non-sensitive
q2	Non-sensitive
q2	Non-sensitive
(*b*) 2-*diverse data using global generalization*	
QID	Disease
Q	HIV
Q	HIV
Q	Non-sensitive
Q	Non-sensitive
Q	Non-sensitive
Q	Non-sensitive
Q	Non-sensitive
(*c*) *External public database*	
Name	QID
Andre	q1
Kim	q1
Jeremy	q2
Victoria	q2
Ellen	q2
Sally	q2
Ben	q2

Based on this minimality principle, an adversary, who knows what mechanism and algorithm employed to anonymize the data, can launch a privacy attack. Wong et al. demonstrated this attack with the following example.

Example (minimality attack) [10]: Assume the data in Table 2a is anonymized such that the number of distinct sensitive values in every equivalence class is at least two in order to protect the sensitive values of individuals with quasi identifier value of q1. This is actually the goal of l-diversity model with $l = 2$ which will be discussed in the next section. This anonymized data is shown in Table 2b. Assume an adversary who knows that l-diversity algorithm is employed to make the data 2-diverse. Also she knows that a global generalization is used. In addition to this knowledge, the adversary has access to the external data shown in Table 2c which is mapped to the same set of individuals in Table 2. Since both quasi identifier values q1 and q2 are generalized to a general value Q (Table 2b), the adversary

concludes that there was at least one equivalence class in the original data set violating 2-diversity, because otherwise, according to minimality principle, no generalization was needed. In addition, as in Table 2c there are five records with quasi identifier value of q2, the adversary will conclude that the equivalence class corresponding to q2 was not violating 2-diversity. Because even if both records with sensitive value HIV belonged to this group, then there would be three other records with non-sensitive values and, therefore, 2-diversity could be satisfied. Based on this reasoning, the adversary concludes that the equivalence class corresponding to q1, which contain 2 records, was not satisfying 2-diversity and this will lead her to this conclusion that both individuals with the quasi identifier value q1 have HIV, i.e Andre and Kim.

4 *k*-Anonymity

An identity disclosure in relational data can happen when an adversary finds a (or a few) match(es) in the released data set for those quasi identifiers she knows about an individual. To avoid this type of disclosure, several techniques are proposed in the literature [16] among which sampling, swapping values and randomization have been some of the most common approaches. However, in all these techniques the data is disturbed such that the correctness of single records are compromised. Therefore, these techniques are inappropriate in the applications where the "truthfulness" of the released data is required [17]. An alternative approach to deal with this limitation is data anonymization. In this technique, individuals' identifying information is either removed or altered to ensure the anonymity of individuals. The most common approach in data anonymization is the notion of *k-anonymity* [18–20] which was proposed by Samarati and Sweeney.

Definition 1 *k-anonymity* This privacy model not only protects the data against identity disclosure but also preserves the truthfulness of the data. A data set satisfies *k-anonymity* iff for every combination of values of quasi identifiers, there are at least k records in the data set sharing those values. In other words, each record in a *k-anonymous* data set is indistinguishable from at least k-1 other records with respect to a set of quasi identifiers [17–20]. In a k-anonymous data set the probability of linking an individual to a specific record with respect to the values of quasi identifiers is at most $\frac{1}{k}$ [21].

Table 3 shows an example of a 3-anonymous data set where ZIP code, date of birth and nationality are quasi identifiers.

In order to make data k-anonymous, the first step is to recognize the set of quasi identifiers in the data set. Choosing quasi identifiers depends on determining what external sources of information an adversary may have to launch a linking attack [17]. The simplest assumption, which was made in the original k-anonymity [18] and most of its refined versions, is to consider a single quasi identifier consisting of all attributes

Table 3 3-anonymous data

PID	Zip code	Date of birth	Nationality	Disease
1	120**	1967	*	Heart disease
2	120**	1967	*	Bronchitis
3	120**	1967	*	Viral infection
4	120**	1970	*	Viral infection
8	120**	1970	*	Cancer
9	120**	1970	*	Cancer
5	118**	1964	*	Cancer
6	118**	1964	*	Heart disease
7	118**	1964	*	Flu
10	118**	1964	*	Diabetes

that can potentially exist in the external sources and can be employed by an adversary for linking attacks.

Although this assumption provides more protection due to considering more attributes as quasi identifiers, this will lead to more data distortion since the records in an equivalence class must agree on more attributes to satisfy k-anonymity [21]. Fung et al. [22, 23] addressed this problem by considering multiple sets of quasi identifiers. According to their work, the data must be k-anonymous with respect to every set of quasi identifiers. For instance, if there are two sets of quasi identifiers X_{nc_1} and X_{nc_2}, then each record must be indistinguishable from $k-1$ other records with respect to both X_{nc_1} and X_{nc_2}. Those $k-1$ other records can be different for each set of quasi identifiers. The only challenge in applying this technique is that the data publisher needs to know how and based on what information the adversary will do a linking attack. Otherwise, this may cause higher data distortion or more disclosure risks [21].

To enforce k-anonymity to a data set, the original model [18] and most of its improved subsequent versions [15, 24–28] employed *generalization* and *suppression*. These two anonymization techniques, unlike the other approaches like swapping and adding noise, retain the truthfulness of the records in the data, and, therefore, satisfy the main goal of k-anonymity [17].

In most versions of k-anonymity including the original model the assumption was that there is just one record for each individual in a data set. With this assumption, as soon as there are k records for every combination of values of quasi identifiers, k-anonymity is satisfied. However, when there is more than one record for each individual in a data set, those methods fail to protect the data. This is because in some equivalence classes, those k records may correspond to less than k distinct individuals. To address this challenge, Wang and Fung proposed (X, Y)-*anonymity* [29].

Definition 2 *(X, Y)-anonymity* Wang and Fung introduced the notion of (X, Y)-*anonymity* to deal with the cases when more than one record in a data set belongs

to an individual. This model requires that each value on X to be linked to at least k *distinct* values on Y, where X and Y are disjoint sets of attributes. K-anonymity is a special case of (X,Y)-*anonymity* where X is the set of quasi identifiers and Y is a key that uniquely identifies an individual's record, such as *ID* [21]. For instance, consider the data set in Table 3 with attributes *PID*, *ZIP Code*, *Date of Birth*, *Nationality*, and *Disease*. With respect to (X,Y)-*anonymity*, to make the data k-anonymous X must be {*ZIP Code*, *Date of Birth*, *Nationality*} and Y must be *PID* and every value of X must be linked to k distinct values on Y, i.e. k distinct patient IDs. Therefore, each individual will be indistinguishable from $k-1$ other individuals.

The other assumption in most versions of k-anonymity is that a single table needs to be anonymized. However, there are some cases when a database contains multiple relational tables and, therefore, those methods either fail to anonymize that database or incur a high information loss to make the data k-anonymous [30]. To deal with this limitation Nergiz et al proposed the notion of *MultiR k-anonymity* [30] .

Definition 3 *MultiR k-anonymity* In this model, the authors assume a database containing a person specific table PT and a set of tables $T_1, T_2, \ldots, T_n$. PT has an identifier attribute *pid* and some sensitive attributes and each table $T_i, 1 \leq i \leq n$, contains a set of quasi identifiers and sensitive attributes as well as some foreign keys. According to this privacy model, the data is k-anonymous if for each individual o corresponding to the join of all tables $PT \bowtie T_1 \bowtie \cdots \bowtie T_n$, there are at least $k-1$ other individuals who have the same values of quasi identifiers as o [21].

4.1 Microaggregation: k-anonymity for Numerical Data

In order to obtain a k-anonymous data set, microaggregation [31] builds clusters of at least k records and replaces each original record by the centroid of the cluster to which this record belongs. The goal is therefore to find a clustering where each cluster contains at least k points and where the sum of distances between the original points and the centroids of the corresponding clusters is minimized. This problem is NP-hard [32] therefore one usually considers heuristic algorithms for microaggregation, for example CBFS (Centre-Based Fixed-Size) [33].

CBFS works as follows. Firstly, the average record $\bar{x}$ of all records in X is computed, then the most distant record x_r to the average record $\bar{x}$ is considered, and a cluster around x_r is formed, containing x_r together with the $k-1$ closest records to x_r. When this cluster is done, all records belonging to this cluster are removed from X. This process is repeated until all the records are assigned to one cluster. Finally, the protected data set X' is built by replacing each original record in X with the centroid of the cluster to which the record belongs. Formally CBFS algorithm is described in Algorithm 1

Algorithm 1: CBFS

```
   Data: X: original data set, k: integer
   Result: X': protected data set
1  begin
2      while (|X| > (2k − 1)) do
3          Compute the average record x̄ of all records in X;
4          Consider the most distant record x_r to the average record x̄;
5          Form a cluster around x_r;
6          ;   /* The cluster contains x_r together with the k − 1 closest records to x_r */
           Remove these records from data set X;
7      Form a cluster with the remaining records;
8  end
```

5 *p*-Sensitivity *k*-Anonymity

In [34], an evolution of k-anonymity called p-sensitive k-anonymity was presented. Its purpose is to protect against attribute disclosure by requiring that there be at least p different values for each confidential attribute within the records sharing a combination of key attributes. The formal definition is as follows.

Definition 4 A data set is said to satisfy p-sensitive k-anonymity for $k > 1$ and $p \leq k$ if it satisfies k-anonymity and, for each group with the same combination of quasi-identifier values that exists in the data set, the number of distinct values for each confidential attribute is at least p

An attacker trying to obtain the confidential value for a given record that has been linked to the p-sensitive k-anonymous data set will not be able to determine which of the p different values inside the group is the corresponding one. p-Sensitive k-anonymity may cause a huge data utility loss in some data sets. In some cases, p-Sensitive k-anonymity is insufficient to prevent attribute disclosure due to the skewness attack and the similarity attack.

5.1 p-Sensitivity k-Anonymity for Numerical Data

In order to obtain p-Sensitivity k-anonymity for numerical data we can modify the way CBFS algorithm builds the clusters (line 5 of Algorithm 1). There are two possible options:

- ***p*-sensitive first** Add p 'close' records with different confidential values, then add the $k - p$ closest records
- ***k*-anonymity first** Form a cluster with the k closest records, then if p-sensitive does not hold, add enough 'close' records with different confidential values to hold it

In [35], P-sensitive first approach was proven as the most appropriate in terms of within-groups homogeneity and, consequently, of data utility.

6 l-Diversity k-Anonymity

Machanavajjhala et al. [4, 5] showed that there are two types of privacy attacks, namely *homogeneity attack* and *background knowledge attack* described in Sect. 3, that k-anonymity and p-Sensitivity k-Anonymity fail to protect against. As a result, the adversary will be able to infer some sensitive information about the individuals even without identifying their records. These facts are employed by Machanavajjhala et al. and they proposed *l-diversity* principle to overcome the limitations of k-anonymity.

Definition 5 *l-diversity* [4, 5]. A data set is l-diverse, if every equivalence class in this data set has at least l "well represented" values for the sensitive attribute.

A 3-diverse version of the data in Table 1 is shown in Table 4. It is obvious that this data set is not vulnerable to *homogeneity attack* and *background knowledge attack*.

Machanavajjhala et al. [4, 5] presented several instances of l-diversity principle based on the definition of term "well represented". The simplest model requires that the number of distinct values for sensitive attributes in every equivalence class to be at least l. This is equivalent to *p-sensitive k-anonymity* principle introduced by Truta and Bindu [36]. Another principle similar to this variant of l-diversity is (α, k)-*anonymity* [37]. A data set is said to satisfy (α, k)-*anonymity* if it satisfies

Table 4 3-diverse data set [5]

	Non-sensitive			Sensitive
	Zip code	Age	Nationality	Condition
1	1305*	≤40	*	Heart disease
4	1305*	≤40	*	Viral infection
9	1305*	≤40	*	Cancer
10	1305*	≤40	*	Cancer
5	1485*	>40	*	Cancer
6	1485*	>40	*	Heart disease
7	1485*	>40	*	Viral infection
8	1485*	>40	*	Viral infection
2	1306*	≤40	*	Heart disease
3	1306*	≤40	*	Viral infection
11	1306*	≤40	*	Cancer
12	1306*	≤40	*	Cancer

k-anonymity and the probability of inferring sensitive values in every equivalence class is at most α.

Another variant of l-diversity is *entropy l-diversity* that requires the entropy of sensitive attribute in every equivalence class to be at least $\log(l)$ [4, 5]:

$$-\sum_{s \in S} P(EC, s) log(P(EC, s)) \geq \log(l) \tag{1}$$

where S is the domain of values of sensitive attribute and p(EC, s) is the fraction of records in equivalence class EC that have value s for the sensitive attribute. For example the entropy of attribute *Condition* in any equivalence class in Table 4 is $-\frac{1}{4}\log(\frac{1}{4}) - \frac{1}{4}\log(\frac{1}{4}) - \frac{2}{4}\log(\frac{2}{4}) = log(2.8)$. Therefore this table satisfies entropy 2.8-diversity. This model was first introduced in [38] as a way of protecting against the homogeneity problem without respect to the role of background knowledge [5].

Entropy l-diversity is motivated by the fact that when the frequency of sensitive values becomes more uniform, then the entropy of sensitive attribute increases [5]. Therefore, by setting a large threshold l, Eq. 1 will be satisfied if the frequency of sensitive attribute is close enough to make the entropy higher than $\log(l)$. However, as the authors in [4, 5] showed, entropy l-diversity will be only possible if the entropy of the sensitive attribute in the entire data set is at least $\log(l)$. This constraint, however, may be too restrictive, particularly when a few values of the sensitive attribute are too frequent, for instance, when 90 % of patients in a data set have heart disease [4, 5]. In this case the entropy of sensitive attribute in the entire data set will be small and therefore only for a small value of l entropy l-diversity can be satisfied. The other shortcoming of entropy l-diversity is that it cannot be easily adopted to define different levels of protection in the cases that sensitive values have different levels of sensitivity[21].

Another notion of l-diversity is *recursive (c, l)-diversity* [4, 5] which mostly focuses on the role of background knowledge of the adversary. Assuming m_i is the number of sensitive values in the equivalence class i, a data set satisfies recursive (c, l)-diversity if in every equivalence class i the frequency of the most frequent sensitive value is less than the sum of the frequencies of the $m_i - l + 1$ least frequent sensitive values multiplied by some constant c. That is, if r_j denotes the number of times the j-th most frequent sensitive value appears in equivalence class i and c is a pre-defined constant, equivalence class i satisfies recursive (c, l)-diversity if $r_1 < c(r_l + r_{l+1} + \cdots + r_{m_i})$. In other words, an equivalence class has recursive (c, l)-diversity if we eliminate one possible value of sensitive attribute and the equivalence class still satisfies (c, l)-diversity [4, 5]. This criterion guarantees that the most frequent sensitive value does not appear too often and the less frequent values do not appear too rarely [21]. However, as it is pointed out in [5], recursive (c, l)-diversity can also be too restrictive.

There is another variant of l-diversity, called *positive disclosure-recursive (c, l)-diversity*, proposed in [4, 5] to deal with the cases that some positive disclosures are acceptable, i.e. when some values of sensitive attribute have less degrees of sensitivity and need not be kept private. The authors define a set Y, called *don't-care set*, which

contains those sensitive values that have minimal sensitivity and positive disclosure of them is allowed. For example, in a context *flu* may be in set Y but *colon cancer* cannot be. Too frequent sensitive values may also be added to Y, for instance when most of the patients visiting a clinic have heart problems then positive disclosure of value *heart disease* may be allowed by the clinic [4, 5]. Having set Y, data will be anonymized to protect just those sensitive values which are not in Y.

This version of *l*-diversity addresses two of the criticisms on *l*-diversity introduced in [11]. The first issue brought up in [11] was that *l*-diversity may incur an excessive level of anonymization. In other words, there may be some cases where there is no need to enforce *l*-diversity. To illustrate this problem, assume the data set of 10,000 patients, considered in Sect. 3 for skewness attack, i.e the data set with one sensitive attribute for test result of a virus with two possible values "*Positive*" and "*Negative*". Obviously a negative test result in this case has low sensitivity and a patient will not mind to be identified with a negative result. But, on the other hand, a patient with positive result is very concerned of being known as a positive case. Therefore, achieving *2-diversity* in this data set will be unnecessary in those equivalence classes which only have "*Negative*" cases. To achieve *2-diversity*, at most $10{,}000 \times 1\% = 100$ equivalence classes from 10,000 records can be built and this will obviously lead to a high information loss. *Positive disclosure-recursive (c, l)-diversity* will not anonymize the data unnecessarily in such cases.

6.1 l-Diversity k-Anonymity for Numerical Data

As we have explained for *p*-sensitivity *k*-anonymity, we can modify the way CBFS algorithm builds the clusters (line 5 of Algorithm 1) to also obtain *l*-Sensitivity *k*-anonymity for numerical data:

- Before adding a new 'close' record, we have to check that the inequality

$$-\sum_{s \in S} P(EC, s) log(P(EC, s)) \geq \log(l)$$

still holds wihtin the cluster.

In some cases we have to remove records or to built very big clusters decreasing in this way the data utility of the protected data.

6.2 l-Diversity k-Anonymity Drawbacks

One criticism on *l*-diversity that is introduced in [11] is that *l*-diversity fails to protect the data against *skewness attack*, described in Sect. 3. To illustrate this drawback, consider the data set of 10,000 patients with positive and negative test results. An

equivalence class that has equal number of positive and negative cases will satisfy entropy 2-diversity and recursive (c, 2)-diversity. However, this equivalence class is susceptible to *skewness attack*.

As another example, consider two equivalence classes so that the first one has 49 positive cases and 1 negative and the second one has 49 negative and only 1 positive cases. Both equivalence classes will be 2-diverse. Also they will satisfy entropy l-diversity for any $l < 1.05$. However, in the first equivalence class with 49 positive test result, probability of considering a patient as a positive case is 98 % while in the second one this probability is only 2 %. But they are dealt with in the same way without considering the issues with skewness of data [11]. On the other hand *positive disclosure-recursive (c, l)-diversity* deals with high sensitive values differently from less sensitive values in *don't-care* set. Hence it can easily address the issues with skewed data mentioned above [7].

7 *t*-closeness

l-diversity does not take into account the semantic relation among sensitive values. Therefore, as it was shown in [11], l-diversity fails to protect the data against *similarity attack*. To overcome the limitations of l-diversity in protecting the data against *skewness attack* and *similarity attack*, Li et al. [11] proposed *t-closeness* privacy model as an extension of l-diversity.

Definition 6 A data set has t-closeness if the distance between the distribution of the sensitive attribute in every equivalence class and the whole data set is at most t. To calculate the distance between distributions the authors use the *Earth Mover Distance* (*EMD*) metric [39].

This privacy model guarantees that the overall distribution will not be skewed in an equivalence class and therefore skewness attack cannot be successful. Also, as the distribution of the sensitive attribute in every equivalence class is almost the same as whole data set, it is unlikely that all values in one equivalence class to be semantically similar.

The limitations of t-closeness are shown in several works. Domingo-Ferrer et al. [40] argued that enforcing almost the same distribution for the sensitive attribute in every equivalence class as the whole data set damages the correlations between quasi identifiers and the sensitive attribute(s), and makes the data useless for analysis[40]. Frikken and Zhang [41] showed that t-closeness can not deal with the situations where some values of a sensitive attribute has more sensitivity than other values and they prosposed (α_i, β_i)-closeness to address this problem. Their main idea was to assign a range to each sensitive value s_i in the domain of the sensitive attribute. An equivalence class then satisfies (α_i, β_i)-closeness if the number of records in the equivalence class having sensitive value s_i is in the range (α_i, β_i). Another drawback of t-closeness, brought up by Li et al. [12], is that *EMD* measure is not an appropriate measure to prevent disclosure of numerical sensitive attributes.

8 *p*-Indistinguishability

As we have explained before, the main idea behind k-anonymity and most of its variants is to group individuals so that any identification is only to a group of k, not to an individual. To do that it is necessary to use the notion of *quasi-identifier*, as defined in the Introduction, a quasi-identifier is any attribute that can be used for an intruder to link a record to a concrete individual. If we assume that sensitive information is not the same for all k records (p-sensitivity or l-diversity concepts ensure this by definition) this throws uncertainly into any knowledge about any individual of the group. In other words, the uncertainly lowers the risk that the intruder's knowledge constitutes an intrusion.

The concept that group knowledge does not violate the privacy of individuals has a long history. National statistical agencies have used this approach to publish aggregated values in the form of contingency tables reflecting the count of individuals holding a particular criterion.

However, in this scenario a new disclosure risk arises within the cells storing only a single (or few) individual(s). The disclosure problem is that combining this data with small cells with other tables may reveal confidential information applying to a single individual. This problem is even worse when data mining or machine learning methods are executed on the top of this aggregation data.

The general question that one should wonder is 'How does it apply to privacy-preserving data mining?' Basically, if we can ensure that disclosures from the data mining algorithms generalize to large enough groups of individuals, then the size of the group can be used as a metric for privacy as in the case of k-anonymity. For a single data mining algorithm this can be easily achieved, for instance, we can prune a decision tree to ensure that decision rules includes enough individuals, etc…However, an still unsolved problem is the cumulative effect of multiple data miming algorithms disclosures. While building a unique model may meet the required privacy requirements (k-anonymity for example), the combination of multiple data mining models may enable deducing individual information

In order to deal with this problem, a metric introduced in [42, 43] uses the concept of anonymity, but specifically focused on the ability to distinguish individuals:

Definition 7 Two records belonging to different individuals I_1 and I_2 are p-indistinguishable given a data set X if for every polynomial time function $f : I \rightarrow \{0, 1\}$

$$|Pr\{f(I_1) = 1|X\} - Pr\{f(I_2) = 1|X\}| \leq p$$

where $0 < p < 1$.

The idea behind the p-indistinguishable concept is that we should be unable to learn a classifier that distinguishes between two individuals with a high probability. Note that, this definition does not prevent us from learning confidential information, it only poses a problem if that confidential information is tied more

closely to one individual rather than another. The main difference of this metric with k-anonymity is that this metric applies to the confidential information of X instead of the quasi-identifiers.

9 Differential Privacy

Differential privacy [44, 45] has emerged in the last years as a rigorous theoretical privacy model to analyze data release methods $\rho(X)$. In particular, and in contrast to what happens in the privacy preservation techniques that we have described in the previous section, there is no assumption about the knowledge of an adversary who wants to attack privacy. The intuitive idea is that a person who allows his personal data to be included in the original data set X must not be worried about his privacy, because the output of $\rho(X)$ looks the same in the case that his data is included in X as in the case that it is not. More formally, a method ρ provides ϵ-differential privacy if for all possible inputs X_1, X_2 that differ in at most one record, and all possible outputs X' of the method ρ, we have $\Pr[\rho(X_1) = X'] \leq e^{\epsilon} \cdot \Pr[\rho(X_2) = X']$, where the probabilities are taken over the randomness of the method ρ.

Most of the papers proposing differentially private methods focus on the scenario where the future clients of the released data are interested in computing a specific family of (few) functions of the initial data. For each such function f, a solution is then to compute the correct value $f(X)$ and then perturb this output, for example by adding Laplace noise. However, in the general scenario that we have considered in this survey, the data owner does not know what or how many functions of X will be computed in the future by the clients who observe the released data set $X' = \rho(X)$. Furthermore, the output X' must have the same structure and size than the input X. The standard solution of adding Laplace noise should be implemented in this way by considering as f the identity function, but then the parameter for the Laplace distribution that ensures differential privacy is so big that the resulting perturbed data is statistically useless.

Some papers [46, 47] have proposed specific methods to achieve differential privacy in the (non-interactive) scenario where the size of the protected data set $X' = \rho(X)$ is the same as the size of the data set X. However, the employed techniques (like generalization and suppression) come from the data-mining world, and the obtained results are tested in relation to data-mining operations, like classification analysis. The obtained utility results for classification are good, but statistical functions are very sensitive to generalization and suppression. This is what motivated the introduction, study and implementation of other protection methods, specific for statistical data and analysis, like rank swapping, shuffling, which are widely accepted and used nowadays by statistical agencies and companies. Unfortunately, these methods can never achieve the notion of differential privacy.

Acknowledgments This work is partially supported by the Ministry of Science and Technology of Spain under contract TIN2012-34557 and by the BSC-CNS Severo Ochoa program (SEV-2011-00067). The authors also acknowledge the support of the Natural Sciences and Engineering Research Council of Canada for this work.

References

1. Domingo-Ferrer, J., Torra, V.: Disclosure control methods and information loss for microdata. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 91–110 (2001)
2. Mateo-Sanz, J.M., Domingo-Ferrer, J., Sebé, F.: Probabilistic information loss measures in confidentiality protection of continuous microdata. Data Min. Knowl. Disc. **11**(2), 181–193 (2005)
3. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N.: Fast data anonymization with low information loss. In: Proceedings of the 33rd International Conference Very Large Data Bases, pp. 758–769 (2007)
4. Kifer, D., Gehrke, J.: l-diversity: privacy beyond k-anonymity. In: Proceedings of IEEE International Conference on Data Engineering (2006)
5. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: l-diversity: privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data, **1**, (2007)
6. Martin, D.J., Kifer, D., Machanavajjhala, A., Gehrke, J., Halpern, J.Y.: Worst-case background knowledge for privacy-preserving data publishing. In: IEEE 23rd International Conference on Data Engineering, pp. 126–135 (2007)
7. Chen, B., Kifer, D., LeFevre, K., Machanavajjhala, A.: Privacy-preserving data publishing. Found. Trends Databases **2**(1–2), 1–167 (2009)
8. Chen, B., LeFevre, K., Ramakrishnan, R.: Privacy skyline: privacy with multidimensional adversarial knowledge. In: VLDB '07 Proceedings of the 33rd international conference on Very large data bases, pp. 770–781 (2007)
9. Li, T., Li, N.: Injector: mining background knowledge for data anonymization. In: ICDE '08 Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, pp. 446–455 (2008)
10. Wong, R.C.-W., Fu, A.W.-C., Wang, K., Pei, J.: Minimality attack in privacy preserving data publishing. In: VLDB '07 Proceedings of the 33rd International Conference on Very Large Data Bases, pp. 543–554 (2007)
11. Li, N., Li, T.: t-closeness: Privacy beyond k-anonymity and l-diversity. In: Proceedings of IEEE International Conference on Data Engineering (2007)
12. Li, J., Tao, Y., Xiao, X.: Preservation of proximity privacy in publishing numerical sensitive data. In: SIGMOD '08 Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 473–486 (2008)
13. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Workload-aware anonymization. In: KDD '06 Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 277–286 (2006)
14. Zhang, Q., Koudas, N., Srivastava, D., Yu. T.: Aggregate query answering on anonymized tables. In: ICDE 2007 Proceedings of the 23rd International Conference on Data Engineering, pp. 116–125 (2007)
15. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Incognito: efficient full-domain k-anonymity. In: SIGMOD '05 Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 49–60 (2005)
16. Ciriani, V., di Vimercati, S.D.C., Foresti, S., Samarati, P.: Microdata protection. In: Yu T., Jajodia S. (eds.) Secure Data Management in Decentralized Systems, pp. 291–321. Springer, New York (2007)

17. Ciriani, V., di Vimercati, S.D.C., Foresti, S., Samarati, P.: k-anonymity. In: Yu T., Jajodia S. (eds.) Secure Data Management in Decentralized Systems, pp. 323–353. Springer, New York (2007)
18. Samarati, P.: Protecting respondents' identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)
19. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report, Computer Science Laboratory, SRI International (1998)
20. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. **10**(5), 557–570 (2002)
21. Fung, B.C.M., Wang, K., Chen, R., Yu, P.S.: Privacy-preserving data publishing: a survey on recent developments. ACM Comput. Surv. (CSUR). **42**(4), (2010)
22. Fung, B.C.M., Wang, K., Yu, P.S.: Top-down specialization for information and privacy preservation. In ICDE '05: Proceedings of the 21st International Conference on Data Engineering, pp. 205–216 (2005)
23. Fung, B.C.M., Wang, K., Yu, P.S.: Anonymizing classification data for privacy preservation. IEEE Trans. Knowl. Data Eng. **19**(5), 711–725 (2007)
24. Bayardo, R.J., Agrawal, R.: Data privacy through optimal k-anonymization. In: ICDE '05 Proceedings of the 21st International Conference on Data Engineering, pp. 217–228 (2005)
25. El Emam, K., Dankar, F.K., et al.: A globally optimal k-anonymity method for the de-identification of health data. JAMIA **16**, 670–682 (2009)
26. Iyengar, V.S.: Transforming data to satisfy privacy constraints. In: KDD '02 Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 279–288 (2002)
27. Sweeney, L.: Achieving k-anonymity privacy protection using generalization and suppression. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. **10**(5), 571–588 (2002)
28. Winkler, W.: Using simulated annealing for k-anonymity. Technical Report 7, U.S. Census Bureau (2002)
29. Wang, K., Fung, B.C.M.: Anonymizing sequential releases. In: KDD '06 Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 414–423 (2006)
30. Nergiz, M.E., Clifton, C., Nergiz, A.E.: Multirelational k-anonymity. IEEE Trans. on Knowl. Data Eng. **21**(8), 1104–1117 (2009)
31. Defays, D., Anwar, M.: Micro-aggregation: a generic method. In: Proceedings of the 2nd International Seminar on Statistical Confidentiality, pp. 69–78 (1995)
32. Oganian, A., Domingo-Ferrer, J.: On the complexity of optimal microaggregation for statistical disclosure control. Stat. J. United Nations Econ. Comm. Eur. **18**(4), 345–354 (2000)
33. Laszlo, M., Mukherjee, S.: Minimum spanning tree partitioning algorithm for microaggregation. IEEE Trans. Knowl. Data Eng. **17**(7), 902–911 (2005)
34. Truta, T.M., Vinay, B.: Privacy protection: p-sensitive k-anonymity property. In: 2nd International Workshop on Private Data Management PDM. IEEE Press (2006)
35. Domingo-Ferrer, J., Sebé, F., Solanas, A.: Microaggregation heuristics for p-sensitive k-anonymity. In: UNECE work session statistical data confidentiality (2008)
36. Truta, T.M., Vinay, B.: Privacy protection: p-sensitive k-anonymity property. In: Proceedings of the 22nd International Conference on Data Engineering Workshops, p. 94 (2006)
37. Wong, R., Li, J., Fu, A., Wang, K.: (α, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. In: In ACM SIGKDD, pp. 754–759 (2006)
38. Ohrn, A., Ohno-Machado, L.: Using Boolean reasoning to anonymize databases. Artif. Intell. Med. **15**(3), 235–254 (1999)
39. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. Comput. Vis. **40**(2), 99–121 (2000)
40. Domingo-Ferrer, J., Torra, V.: A critique of k-anonymity and some of its enhancements. In: ARES '08 Proceedings of the 2008 Third International Conference on Availability, Reliability and Security, pp. 990–993 (2008)

41. Frikken, K.B., Zhang, Y.: Yet another privacy metric for publishing micro-data. In: WPES '08 Proceedings of the 7th ACM workshop on Privacy in the electronic society, ACM, pp. 117–122 (2008)
42. Clifton, C., Kantarcioglu, M., Vaidya, J.: Defining privacy for data mining. In: Proceedings of National Science Foundation Workshop on Next Generation Data Mining (2002)
43. Vaidya, J., Clifton, C., Zhu, M.: Privacy Preserving Data Mining. Springer, New York (2006)
44. Dwork, C.: Differential privacy. In: International Colloquium on Automata, Languages and Programming, volume 4052 of Lecture Notes in Computer Science, pp. 1–12. Springer, New York (2006)
45. Dwork, C.: A firm foundation for private data analysis. Commun. ACM **54**(1), 86–95 (2011)
46. Machanavajjhala, A., Gehrke, J., Götz, M.: Data publishing against realistic adversaries. Proc. Very Large Databases Conf. **2**(1), 790–801 (2009)
47. Mohammed, N., Chen, R., Fung, B.C.M., Yu, P.S.: Differentially private data release for data mining. In: ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 493–501 (2011)

Data Privacy with *R*

Daniel Abril, Guillermo Navarro-Arribas and Vicenç Torra

Abstract Privacy Preserving Data Mining (PPDM) is an application field, which is becoming very relevant. Its goal is the study of new mechanisms which allow the dissemination of confidential data for data mining tasks while preserving individual private information. Additionally, due to the relevance of *R* language in the statistics and data mining communities, it is undoubtedly a good environment to research, develop and test privacy techniques aimed to data mining. In this chapter we outline some helpful tools in *R* to introduce readers to that field, so that we present several PPDM protection techniques as well as their information loss and disclosure risk evaluation process and outline some tools in *R* to help to introduce practitioners to this field.

1 Introduction

R is a free software environment for statistical computing, which has gained a lot of popularity recently. It has become very popular not only for statistics but specially for data mining and machine learning tasks. In this chapter we introduce the reader to the use of R a tool for data privacy research and practice. In this line we not only

D. Abril (✉) · V. Torra
Institut d'Investigació en Intel·ligència Artificial,
Consejo Superior de Investigaciones Científicas Campus de la UAB,
08193 Bellaterra, Catalonia, Spain
e-mail: dabril@iiia.csic.es

D. Abril
UAB, Universitat Autónoma de Barcelona, Barcelona, Spain

V. Torra
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

G. Navarro-Arribas
Department of Information and Communications Engineering,
Universitat Autonoma de Barcelona, 08193 Bellaterra, Catalonia, Spain
e-mail: gnavarro@deic.uab.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_5

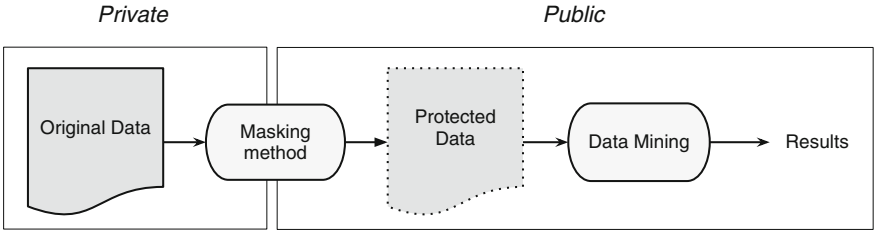


Fig. 1 Data-driven PPDM

review common data privacy tasks in R but also introduce more novel approaches that present current research trends.

In the lines of this work we consider Privacy Preserving Data Mining (PPDM) [6] and Statistical Disclosure Control (SDC) [36] as two disciplines focused towards the same objective. They seek to ensure that data can be published without giving away confidential information that can be linked to specific respondents and also achieve it with the minimum loss of detail. So, third parties can analyze, find patterns, or build classification models with the modified data as working with the original ones. Figure 1 shows the typical scenario, where an original confidential data set is transformed into a protected data set. *Masking methods*, such as perturbation or manipulation, are used to obtain a protected version of the original data, which ensures a given degree of privacy or anonymity. Data mining techniques can be performed on this obtained protected data set by researchers or statisticians without risking the disclosure of sensitive information.

Another important issue in PPDM and SDC is to be able to evaluate the protection provided by a given masking method. The evaluation of a given method has to consider the degree of privacy obtained and at the same time, the perturbation introduced which yields information loss in the protected data as compared to the original one. We describe in this chapter how to compute typical measures of disclosure risk and information loss, and how *R* can also be used to compute more advanced disclosure risk measures formalized as optimization problems. As this chapter is an introduction of some R tools for some basic privacy preserving tasks, it should be desirable that readers have a minimum background of that statistical software.

This chapter is organized as follows. Section 2 describes the kind of data used in the stated problem. Section 3 introduces and classifies a set of different techniques frequently used to protect data. In Sect. 4 are presented the standard metrics to evaluate the protected data. Section 5 introduces a deeper study about the re-identification risk, presenting complex techniques relying on optimization problems in order to help us to evaluate the disclosure risk more accurately. Finally, Sect. 6 concludes the chapter.

2 Microdata

In SDC and PPDM we usually work with *microdata* files. A *microdata* file is a set of records containing information on some given entities (such as individuals or companies). The information for each entity is expressed in terms of some given attributes or variables. So, any of these files can be seen as a matrix with n rows (*records*) and V columns (*attributes*), where each row refers to a single individual or entity. We comment that in the whole chapter we use the words attributes and variables making reference to the same concept, the data columns.

Among the different types of data attributes that can be found in the files, the most typical ones are the numerical and categorical (either nominal or ordinal). Nevertheless, other types of data can also be found. Some of them are time series [22], Web server logs for usage mining [20], or query logs for user profiling [21]. Table 1 is an example of a microdata with 12 individuals (records) and 8 numerical attributes.

The attributes in a dataset can be classified in two different categories, depending on their capability to identify unique individuals, as follows:

- *Identifiers*: attributes that can be used to identify the individual unambiguously. A typical example of identifier is the passport number.
- *Quasi-identifiers*: attributes that are not able to identify a single individual when they are used alone. However, when combining several quasi-identifier attributes, they can unequivocally identify an individual [29]. Among the quasi-identifier attributes, we distinguish between confidential and non-confidential, depending on the kind of information that they contain. The ZIP code is an example of a non-confidential quasi-identifier attribute, and the salary is a confidential quasi-identifier. Note that, in general, all attributes are potentially quasi-identifiers.

Protection methods will normally remove or encrypt identifiers, and mask (distort) quasi-identifiers to avoid potential re-identification. In some cases confidential attributes are not masked by the method to preserve utility.

In Table 1 the first attribute, V_0 is an identifier, and all the others can be considered as quasi-identifiers, in what follows we will discuss the masking of attributes $V_1, \ldots, V_7$.

R is very convenient to work with microdata files. They can be represented by a *data frame*, and easily read from a file using the `read.table()` function. Moreover, *R* provides convenient functions to manipulate, read, and save these type of data in the most common formats. See [25] for more information on data management in *R*. Below we provide the corresponding code to load the microdata sample shown in Table 1, which will be used many times in the rest of the chapter.

```
mdata <- as.table(rbind(c(15, 23, 42.01, 23, 50, 1150, 37), c
    (12, 43, 59.93, 28, 70, 1960, 37), c(64,  229, 319.27,12,
   84, 1008, 25), c(12, 45, 62.07, 29, 73, 2117, 30), c(28, 39,
    74.21, 9, 30, 270, 40), c(71, 102, 191.5, 10, 63, 630, 20),
    c(23, 64, 95.16, 9, 74, 666, 10), c(25, 102, 138.14, 72,
   30, 2160, 80), c(48, 230, 301.78, 26, 30, 780, 35), c(32,
   50, 90.62, 6, 45, 270, 15), c(90, 200, 318.4, 8, 45, 360,
   15), c(16, 100, 125.56, 34, 55, 1870, 45)))
```

Table 1 A sample microdata file

Id	Exp 16 %	Exp 7 %	Total	H. paid for	W. rate	W. sum	Total h.
V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7
1	15.00	23.00	42.01	23.00	50.00	1150.00	37.00
2	12.00	43.00	59.93	28.00	70.00	1960.00	37.00
3	64.00	229.00	319.27	12.00	84.00	1008.00	25.00
4	12.00	45.00	62.07	29.00	73.00	2117.00	30.00
5	28.00	39.00	74.21	9.00	30.00	270.00	40.00
6	71.00	102.00	191.50	10.00	63.00	630.00	20.00
7	23.00	64.00	95.16	9.00	74.00	666.00	10.00
8	25.00	102.00	138.14	72.00	30.00	2160.00	80.00
9	48.00	230.00	301.78	26.00	30.00	780.00	35.00
10	32.00	50.00	90.62	6.00	45.00	270.00	15.00
11	90.00	200.00	318.40	8.00	45.00	360.00	15.00
12	16.00	100.00	125.56	34.00	55.00	1870.00	45.00

The file is described in terms of variables $V_1, \ldots, V_7$, which stand for Expenditure at 16 %, Expenditure at 7 %, Total Expenditure, Hours paid for, Wage rate, Wage sum, Total hours.

3 Masking Methods

For a classification of masking methods for data privacy see [35]. In this chapter we will focus on data-driven methods, which do not care about the intended use of the protected data. They usually adopt perturbative approaches and are intended as generic protection methods. Three of the most popular data-driven methods are: additive noise, microaggregation, and rank swapping.

Most of the currently protection methods are implemented in the *sdcMicro* [31] package, a specific *R* package used for the generation of anonymized (micro) data. In addition, various risk estimation methods are also included.

Given the fact that you have already installed *R* one possibility of installing the *sdcMicro* package plus all required packages from a CRAN server, and then load the package is by:

```
install.packages("sdcMicro", depend=TRUE)   #Install
library(sdcMicro)                           #Load
```

3.1 Additive Noise

Additive Noise consists of adding random noise with or without the same correlation structure as the original unmasked data [7, 11]. A simple example of noise addition is introducing noise according to a normal distribution $N(0, p\sigma)$, where σ is the

standard deviation of the original data, and p is the parameter of the method. This can be computed with the following *R* code, using the `rnorm` function, a normal distribution function.

```
p <- 0.2
datmask <- apply(mdata, 2, function(x) { x + rnorm(dim(mdata)[1],
    0, p*sd(x))})
```

Note that we are protecting the variable *mdata*, which corresponds to the small microdata example introduced in Sect. 2. Nevertheless, the previous code can be easily reproduced with the methods included in the *sdcMicro* package. This package provides `addNoise()`, a powerful method to anonymize data through several configurations of Additive Noise. Using the following commands we obtain the same resulting masked data than with the previous command (result in Table 2a).

```
datmask <- addNoise(mdata, noise=0.2, method="additive")$xm
```

Note that the function *addNoise* has a parameter, *method*, to indicate the type of noise method we are going to apply. We can use simple methods such as *additive*, which adds noise completely at random to each variable depending on their size and standard deviation, but there are much complex methods such as, *correlated* and *correlated2*, which add noise and preserve the covariance.

3.2 Microaggregation

Microaggregation is a masking method, which clusters data into small clusters and then replaces the original data by the centroids of the corresponding clusters. Privacy is ensured by requiring each cluster to have at least k elements, satisfying k-anonymity [27, 28]. Microaggregation was originally defined in [8] for numerical attributes, but later extended to other domains such as categorical data in [32] (see also [13]), constrained domains in [33] and vector spaces [4].

As the solution of optimal microaggregation is NP-Hard [23] when we consider more than one variable at a time (multivariate microaggregation), some heuristic methods have been developed. MDAV [9] (Maximum Distance to Average Vector) is one of such existing algorithms.

`microaggregation()` is the corresponding function provided by the *sdcMicro* package for various Microaggregation methods. For example, in order to perform the heuristic microaggregation method MDAV with at least 3 records per cluster (aggregation level) we run the following command.

```
res <- microaggregation(mdata, method="mdav", aggr=3)$mx
```

Table 2b shows the masked data resulting from the microaggregation of the original data (Table 1). Note the 4 clusters created satisfying k-anonymity [27, 28]. A protected dataset is said to satisfy k-anonymity if each combination of quasi-identifiers is shared by at least k records. That is, there are at least k indistinguishable entities (considering their quasi-identifiers) in the protected data set.

Table 2 Masked microdata

(a) Noise Addition using a normal distribution

V_1	V_2	V_3	V_4	V_5	V_6	V_7
29.33	42.19	28.15	25.42	49.20	1145.09	35.64
9.18	50.36	55.79	30.07	67.44	2025.94	36.90
60.50	228.62	315.35	13.07	84.97	1088.69	24.57
11.82	63.78	38.07	28.20	72.06	2280.62	25.82
28.22	8.41	100.15	8.00	31.82	119.88	42.57
73.27	137.69	186.59	13.56	63.99	521.14	17.07
22.19	57.88	97.94	15.42	66.56	661.83	6.95
30.22	93.87	96.69	65.04	33.53	2144.85	78.23
47.25	217.32	328.05	25.06	31.55	831.52	31.20
34.92	58.27	98.20	4.77	44.25	515.82	18.24
81.42	173.66	307.26	2.73	41.87	333.38	14.37
12.92	99.61	117.77	39.66	49.88	1708.59	46.41

(b) Microaggregation

V_1	V_2	V_3	V_4	V_5	V_6	V_7
18.33	35.67	59.43	20.33	51.00	1179.00	35.67
17.67	81.67	107.88	44.67	51.67	1996.67	54.00
39.67	114.33	168.35	9.00	67.67	648.00	16.67
18.33	35.67	59.43	20.33	51.00	1179.00	35.67
18.33	35.67	59.43	20.33	51.00	1179.00	35.67
69.67	177.33	270.56	14.67	46.00	590.00	23.33
39.67	114.33	168.35	9.00	67.67	648.00	16.67
17.67	81.67	107.88	44.67	51.67	1996.67	54.00
69.67	177.33	270.56	14.67	46.00	590.00	23.33
39.67	114.33	168.35	9.00	67.67	648.00	16.67
69.67	177.33	270.56	14.67	46.00	590.00	23.33
17.67	81.67	107.88	44.67	51.67	1996.67	54.00

(c) Rank Swapping

V_1	V_2	V_3	V_4	V_5	V_6	V_7
16.00	50.00	59.93	29.00	70.00	1870.00	30.00
12.00	45.00	42.01	12.00	50.00	2117.00	45.00
71.00	200.00	318.40	28.00	63.00	2160.00	15.00
12.00	43.00	138.14	23.00	74.00	1960.00	37.00
23.00	64.00	125.56	6.00	30.00	360.00	80.00
64.00	230.00	95.16	26.00	84.00	666.00	35.00
28.00	39.00	191.50	8.00	73.00	630.00	15.00
48.00	100.00	62.07	34.00	30.00	1008.00	40.00
25.00	102.00	90.62	10.00	45.00	270.00	20.00

(contuined)

Table 2 (contuined)

90.00	23.00	301.78	9.00	30.00	780.00	25.00
32.00	229.00	319.27	9.00	55.00	270.00	10.00
15.00	102.00	74.21	72.00	45.00	1150.00	37.00

3.3 Rank Swapping

Rank swapping is another of the most popular perturbative methods. In this case, values from the original data are randomly exchanged between records. All the record's values of a variable V_i are ranked in ascending order; then each ranked value of V_i is swapped with another ranked value randomly chosen within a restricted range (e.g., the rank of two swapped values cannot differ by more than p percent of the total number of records). The method was first described for numerical attributes in [19], although the idea of swapping data was first mentioned in [26].

Rank Swapping was also implemented in the *sdcMicro* package. The function provided by this packages is `rankSwap()` and besides the parameter p we are able to tune others such as $R0$ or $K0$. $R0$ is a factor that preserves the correlation between variables within a certain range based on the given value. We can specify the preservation factor as $R0 = \frac{R1}{R2}$ where $R1$ is the correlation coefficient of the two fields after swapping, and $R2$ is the correlation coefficient of the two fields before swapping. While, $K0$ is a subset-mean preservation factor. It preserves the means before and after Rank Swapping within a range based on the given value. K0 is the subset-mean preservation factor such that $|X_1 - X_2| \leq \frac{2K_0X_1}{\sqrt{(N_S)}}$, where X_1 and X_2 are the subset means of the field before and after swapping, and N_S is the sample size of the subset.

Table 2c shows the masked data obtained when applying Rank Swapping to the sample microdata, Table 1, by means of the following command,

```
res <- rankSwap(mdata,P=40,R0=0)
```

4 Evaluation

The evaluation of a protected data set is expressed in terms of data utility and disclosure risk. The former, data utility, is an evaluation about how much information has been lost in the protection process, so it gives an estimation about the usefulness of the protected data. The latter, disclosure risk, evaluates to what extent confidentiality is ensured.

The optimal evaluation for a protected data set is the one that has the minimum information loss (or the maximum data utility) and the minimum disclosure risk. However, these two magnitudes are in contradiction. For example, when masking is not preformed on a released data set, statisticians can obtain fully accurate computations, but the disclosure of an individual is very likely. On the contrary, we

have to perturb the data as much as possible to reach the maximum protection level. Therefore, a good protected data set is the one that achieves a good trade-off between *information loss* (*IL*) and *disclosure risk* (*DR*). This trade-off is frequently expressed by means of the average of these two magnitudes. So, given the original data X and the protected data X' a score, it is formalized as:

$$Score(X, X') = \frac{IL(X, X') + DR(X, X')}{2}$$

In the following sections we describe the information loss and the disclosure risk metrics and how they can be computed in *R*. In Sect. 4.1 we discuss generic information loss measures as well as an example of a specific measure, and Sect. 4.2 presents the disclosure risk generic measures.

4.1 Information Loss (IL)

Information loss measures the differences between the original and the masked data, as well as the differences between the analyses on the original and masked data. In general, given an analysis or statistics S for a data set X, *IL* can be defined as follows:

$$IL(X, X') = d(S(X), S(X'))$$

where d is a function that measures the divergence between the two analyses or statistics and $X' = masked(X)$.

Naturally, when $X' = X$, we expect $S(X) = S(X')$ and $IL(X, X')$ to be zero. Therefore, any distance function on the outcome of S can be appropriate to measure divergence, and thus they can be used as the function d.

When we know which is the expected use of the data, we would use it as S. In this case, we say that we use specific information loss measures. Otherwise, it is common to use some *standard* statistics as generic information loss measures. Naturally, any pair of functions S and d will lead to different information loss measures. Averages of several information loss measures are also common.

Although we have said that we do not expect $S(X) = S(X')$, this is not always the case. Randomness and local minima caused by different initializations might cause that $S(X) = S(X')$. This issue is of great importance when computing information loss because it can cause that greater distortion on X leads to an apparent decrease of information loss.

As stated above, some examples of generic information loss measures found in the literature are defined in terms of common statistics (means, covariances, correlations) [12]. Other based on the former but using a probabilistic approach can be found in [18]. Alternative measures based on entropy [10], and distances [5] also exist.

For specific data usages, we find e.g. [34], which reviews information loss measures for clustering algorithms, a very common analysis in data mining applications.

4.1.1 Generic Information Loss

A well know generic information loss provided by the `sdcMicro` package is known as IL1s [12]. IL1s measures the distance between the protected records and the original ones. Formally, for each record i:

$$IL1s(X, X') = \frac{1}{VM} \sum_{i=1}^{M} \sum_{j=0}^{V} \frac{|x_{ij} - x'_{ij}|}{\sqrt{2} S_j} \tag{1}$$

where V is the number of attributes, M the number of records, x_{ij} denotes the value of record i for attribute j, and x'_{ij} the same for the protected version, and S_j is the standard deviation of the j-th attribute in the original data.

Given an original data set x and a protected one xm the `sdcMicro` package provides a specific function to compute the information loss based on IL1s (Eq. 1). That is,

```
dUtility(x,xm)
```

Note that although the function name is `dUtility`, it actually returns a measure of the information loss and not the data utility. Utility is usually understood as the inverse of the information loss (the larger the loss, the lesser the utility), so it could raise some misunderstandings. `sdcMicro` also provides more (generic) information loss measures based on other statistics through the `summary.micro` function. In particular, it computes measures of IL based on the difference between means, medians, variances, MAD, …

4.1.2 Specific Information Loss

To illustrate the case of a specific information loss measure, we consider [34] where a data miner applies clustering to the protected data. In this case, we need to study the divergence between the clusters obtained from the original data and the clusters obtained from the masked data. This can be done in *R* using the package *fpc*.

We consider an original file x, its masked file x', the k-means clustering algorithm S, and the Adjusted Rand Index (ARI) [15] as the cluster comparison method d. Although, ARI is a similarity measure yielding a value in the interval $[-1, 1]$ we can easily transform it into a distance yielding values between 0 and 1 as $d(\pi, \pi') = 1 - (1 + AdjustedRandIndex(\pi, \pi'))/2$ where π and π' are two cluster partitions (corresponding to $S(x)$ and $S(x')$. In *R* code is as follows:

```
DistARI <- function(ari){
    return (1-(1+ ari)/2)
}
```

The following code computes the Adjusted Rand Index for a masked file computed from `mdata`, Table 1, using noise addition with a given parameter p. We have the function `ILRand` for computing the Adjusted Rand index for a pair of original and masked files. Then, we have the function `ILRandNoiseAddition` which computes `nTimes` executions of k-means for a masked file using noise addition with parameter `p`. In our implementation we consider several computations of the k-means algorithm for both the original and the protected file. This is so, because as discussed above, randomness and local minima caused by different initializations might cause difficulties in computing the information loss. Note that two different executions of k-means with the same data might lead to different clusters. Therefore, `ILRand(original,original)` might be non zero. To solve this problem, we consider several executions for this pair, and select the minimum.

```
# Install fpc package
install.packages("fpc", depend=TRUE)
# Load fpc package
library(fpc)

ILRand <- function(original, masked) {
  ro <- kmeans(original, 3)
  rm <- kmeans(masked, 3)
  index <- cluster.stats(dist(original), ro[1]$cluster, rm[1]$
      cluster)
 randIndex <- index$corrected.rand
  return (DistARI(randIndex))
}

ILRandNoiseAddition <- function(original){
  function(nTimes){
    function(p){
      masked <- apply(original, 2, function(x){ x + rnorm(dim(
          original)[1], 0, p * sd(x))})
      min(replicate(10, ILRand(original, masked)))
    }
  }
}
```

Using these two functions we can compute and plot (see Fig. 2) the information loss for a large range of p values. That is,

```
# Compute the information for mdata
iLrand <- sapply((1:100) * 0.01, ILRandNoiseAddition(mdata)(10))
# Plot iLrand
plot(iLrand, ylim=c(0, 0.6), xlab="p*100")
```

4.2 Disclosure Risk

Disclosure risk (DR) measures the risk of re-identification, that is the extent in which entities are identified in the protected files (identity disclosure) or the extent in which some information can be inferred for the entities (attribute disclosure).

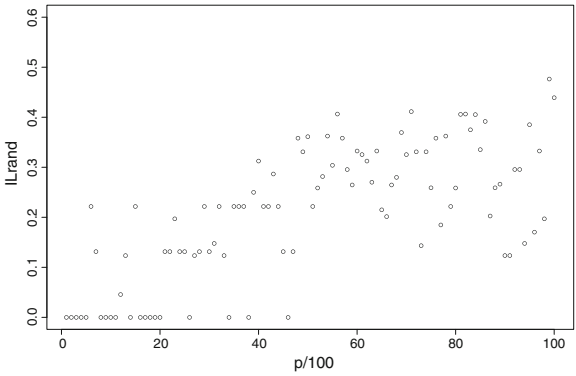


Fig. 2 Cluster-based information loss

A simple way to evaluate attribute disclosure is the interval disclosure measure [12]. In this case we consider an interval around the protected value and check if the original value yields in this interval. The interval can be determined by a percentage of the ranked values or the standard deviation of the corresponding variable. The *sdcMicro* package provides measures for interval disclosure using a robust Mahalanobis distance to determine those intervals [30]. Several variations of this measure can be computed with the function `dRiskRMD`:

```
dRriskRMD(x,x')$risk1
```

Identity disclosure is normally defined as the risk that an intruder having additional knowledge can link a record from the masked file with its corresponding record in an intruder's data file. For the sake of measurability, the original file is often used as intruder's file. This fact is considered as the worst case scenario. That is, the case where an intruder knows who is in the original database, and has information of all the attributes in the database.

Then, given the original (or intruder's) file and its corresponding masked file, it is possible to use record linkage algorithms to establish links between records from both files which correspond to the same entity. There are two main approaches of record linkage: probabilistic record linkage (PRL) [16] and distance-based record linkage (DBRL) [24]. Both approaches have been used extensively in the area of data privacy to evaluate the disclosure risk of protected data.

In the next section we introduce the latest research techniques to obtain better estimations of the disclosure risk.

5 Advanced Techniques for Disclosure Risk Evaluation

Following the idea stated in the previous section about how to estimate the disclosure risk through the worst case scenario we present a new procedure to evaluate the risk. This relies on a supervised learning approach for distance-based record linkage.

The idea is to use a parameterized distance and a supervised approach to learn the parameters that maximize the number of correct links between the records from both files. In this way, we can evaluate the risk with the number of such correct links.

In particular, there is research with parameterized distances using a weighted mean [2], a Choquet integral [3], and a symmetric bilinear function [1]. Here we illustrate the use of *R* with the weighted mean because it is the simplest and gives good results.

In the following sections we define the parametric distance (Sect. 5.1) and the supervised approach in terms of an optimization problem (Sect. 5.2). Both sections use the following notation: $V_1^X, \ldots, V_n^X$ and $V_1^{X'}, \ldots, V_n^{X'}$ denote the set of attributes of file X and X', respectively. File X corresponds to the original file and X' the masked file. Then, a record a in X corresponds to $a = (V_1^X(a), \ldots, V_n^X(a))$ and, similarly, $b = (V_1^{X'}(b), \ldots, V_n^{X'}(b))$. $\overline{V_i^X}$ and $\sigma(V_k(X))$ correspond to the mean and the standard deviation of all values corresponding to the ith attribute from file X.

5.1 A Parametric Distance for Record Linkage

One of the basic points in the approach is that the multiplication of the Euclidean distance by a constant will not change the results of the record linkage algorithm. Due to this, we can replace the Euclidean distance in distance-based record linkage by an arithmetic mean or a weighted mean of the distances for the attributes. To do so, we consider the following distance:

$$d^2(a,b) = \sum_{i=1}^{n} \frac{1}{n} \left(\frac{V_i^X(a) - \overline{V^X}_i(a)}{\sigma(V_i^X)} - \frac{V_i^{X'}(b) - \overline{V^{X'}}_i(b)}{\sigma(V_i^{X'})} \right)^2$$

That, defining

$$d_i^2(a,b) = \left(\frac{V_i^X(a) - \overline{V^X}_i(a)}{\sigma(V_i^X)} - \frac{V_i^{X'}(b) - \overline{V^{X'}}_i(b)}{\sigma(V_i^{X'})} \right)^2$$

we can rewrite as

$$d^2AM(a,b) = AM(d_1^2(a,b), \ldots, d_n^2(a,b)),$$

where AM is the arithmetic mean $AM(c_1, \ldots, c_n) = \sum_i c_i/n$.

From this definition, we can consider a weighted version of the Euclidean distance, defined as follows.

Definition 1 Let $p = (p_1, \ldots, p_n)$ be a weighting vector (i.e., $p_i \geq 0$ and $\sum_i p_i = 1$). Then, the weighted distance is defined as:

$$d^2WM_p(a,b) = WM_p(d_1^2(a,b),\ldots,d_n^2(a,b)),$$

where $WM_p = (c_1,\ldots,c_n) = \sum_i p_i \cdot c_i$.

The interest of this variation is that we do not need to assume that all the attributes are equally important in the re-identification. This would be the case if one of the attributes is a key-attribute, e.g., an attribute where $V_i^X = V_i^{X'}$. In this case, the corresponding weight would be assigned to one, and all the others to zero, leading to a 100 % of re-identifications.

Moreover, this definition permits us to apply a supervised learning approach to determine the parameters of the method according to some fixed constraints. In this way, we can tune the distance to have a better performance.

5.2 Determining the Optimal Parameters

For the sake of simplicity in the formalization of the process, we assume that each record b_i of X' is the protected version of a_i of X. That is, files are aligned. Then, two records are correctly linked when the weighted mean d^2WM_p of the aligned values, $d^2WM_p(a_i,b_i)$, is smaller than the non-aligned values, $d^2WM_p(a_i,b_j)$, for all $i \neq j$. Formally, we have that an a_i record is correctly matched when the following equation holds for all $i \neq j$. That is, for all i and j s.t. $i \neq j$,

$$d^2WM_p(a_i,b_i) < d^2WM_p(a_i,b_j) \tag{2}$$

In optimal conditions these inequalities should be true for all records a_i. Nevertheless, we cannot expect this to hold because of the errors introduced in the data by the protection method. Then, the learning process is formalized as an optimization problem with an objective function and some constraints.

Equation (2) should be relaxed so that the solution can be violated by some pair i, j. The relaxation is based on the concept of blocks. We consider a block as the set of equations concerning record a_i. Therefore, we define a block as the set of all the distances between one record of the original data and all the records of the protected data. Then, we assign to each block a variable K_i. Therefore, we have as many K_i as the number of rows of our original file. Besides, we need for the formalization a constant C that multiplies K_i to overcome the inconsistencies and satisfy the constraint. This variable K_i indicates, for each block, if all the corresponding constraints are accomplished ($K_i = 0$) or not ($K_i = 1$). Then, we want to minimize the number of blocks non compliant with the constraints (i.e., the number of non correctly linked records). This way, we can find the best weights that minimize the number of violations, or in other words, we can find the weights that maximize the number of re-identifications between the original and protected data.

The rationale of this formalization is that if for a record a_i, Eq. (2) is violated for a certain record b_j, then, it does not matter that other records b_j also violate the same equation for the same record a_i. This is so because record a_i will not be re-identified.

Using these variables K_i and the constant C, we have that all pairs $i \neq j$ should satisfy the following equation,

$$d^2WM_p(a_i, b_j) - d^2WM_p(a_i, b_i) + CK_i > 0 \tag{3}$$

As K_i is only 0 or 1, we use the constant C as the factor needed to really overcome the constraint. In fact, the constant C expresses the *minimum distance* we require between the correct link and the other incorrect links. The larger it is, the more correct links are distinguished from incorrect links.

Using these constraints the optimization problem is as follow:

$$Minimize \sum_{i=1}^{N} K_i \tag{4}$$

$$Subject\ to:$$

$$d^2WM_p(a_i, b_j) - d^2WM_p(a_i, b_i) + CK_i > 0 \quad \forall i, j = 1, \ldots, N, i \neq j \tag{5}$$

$$K_i \in \{0, 1\} \tag{6}$$

$$\sum_{i=1}^{n} p_i = 1 \tag{7}$$

$$p_i \geq 0 \tag{8}$$

where N is the number of rows of both data files (original and protected), and n the number of attributes of those files. Note that this problem has $n + N$ variables, and $N * (N - 1) + 1$ constraints.

This is a mixed integer linear problems (MILP), because it is dealing with binary (K_i) and real-valued variables (weights, p_i) in the objective function and in the constraints, respectively. In order to compute the estimation of the disclosure risk of a protected data set we use the *lpSolveAPI R* package [17]. This package provides an *R* interface for the *lp_solve* library, a mixed integer linear programming (MILP) solver with support for pure linear, (mixed) integer/binary, semi-continuous and special ordered sets (SOS) models. The *lp_solve* library uses the revised simplex method to solve pure linear programs and uses the branch-and-bound algorithm to handle integer variables, semi-continuous variables and special ordered sets.

Consider the toy example presented in Table 3. It consists of two small data sets with 3 attributes and 3 records, the original file X (left), and its protected version X' (right). Note that $V_1^X = V_1^{X'}$, so it is expected that the result, or one of the optimal results, must be the weighting vector [1, 0, 0], since the first column is enough to link all the records between both data sets.

Table 4 shows the constraints of this example (according to Eq. (5) above) defined with the constraint C set to 10. In the optimization problem the data is normalized.

As mentioned above, to solve the optimization problems we propose the utilization of the *lpSolveAPI* package, although, there are similar packages which are also able

Table 3 Example data sets

Original (X)			Protected (X')		
1.00	30.00	27.00	1.00	20.00	47.00
2.00	50.00	47.00	2.00	20.00	47.00
3.00	25.00	31.00	3.00	23.00	31.00

Table 4 Matrix of constraints corresponding to the toy example (Table 3)

Constraint's equations	Constraints' Matrix					
Equation (5)	p_1	p_2	p_3	K_1	K_2	K_3
$d^2WM_p(a1,b1) - d^2WM_p(a1,b2) + 10K_1 > 0$	1.00	0.00	0.00	10	0	0
$d^2WM_p(a1,b1) - d^2WM_p(a1,b3) + 10K_1 > 0$	4.00	2.31	−1.62	10	0	0
$d^2WM_p(a2,b2) - d^2WM_p(a2,b1) + 10K_2 > 0$	1.00	0.00	0.00	0	10	0
$d^2WM_p(a2,b2) - d^2WM_p(a2,b3) + 10K_2 > 0$	1.00	−2.93	4.93	0	10	0
$d^2WM_p(a3,b3) - d^2WM_p(a3,b1) + 10K_3 > 0$	4.00	−3.62	0.31	0	0	10
$d^2WM_p(a3,b3) - d^2WM_p(a3,b2) + 10K_3 > 0$	100	−3.62	0.31	0	0	10

to solve this kind of problems, such as [14]. In order to solve our problem the first step is the creation of a *lpSolve linear program model object* (LPMO) with $N*(N-1)+1$ constraints and $n + N$ decision variables, where N is the number of rows and n the number of attributes. In this example, as we have 3 rows and 3 columns for each dataset the values for N and n are both 3. So, the *R* code to create a LPMO with the stated number of constraints is the following:

```
#Install lpSolveAPI
install.packages("lpSolveAPI", depend=TRUE)
#Load lpSolveAPI
library(lpSolveAPI)

nrows <- nrow(original)
ncols <- ncol(original)

lpmo <- make.lp( nrow=(nrows * (nrows - 1) + 1), ncol=(ncols +
    nrows))
```

Once the LPMO is created, the next step is to set the objective function, constraints, and bounds. We create a constraint matrix for Eq. (5) without taking into account the CK_i part. That matrix, with $N * (N - 1)$ rows and n columns, is represented by the middle columns of Table 4: columns p_1, p_2, p_3. At the end of each column we add a 1 value, because the last row represents the constraints expressed by Eq. (7).

```
#Constraints by columns
for(i in 1:(ncols)){
    set.column(lpmo, i, c(constraints[, i], 1))
}
```

Regarding the second part of the constraints, the CK_i part, we define $C = 10$. When data is normalized, 10 is enough to express the minimum distance required between a correct and an incorrect link. As it was said, there are $n + N$ decision

Table 5 Constraints matrix

Model name:						
	p_1	p_2	p_3	K_1	K_2	K_3
Minimize	0	0	0	1	1	1
R1	1	0	0	10	0	$0 \geq 0.0001$
R2	4	2.31	−1.62	10	0	$0 \geq 0.0001$
R3	1	0	0	0	10	$0 \geq 0.0001$
R4	1	−2.93	4.93	0	10	$0 \geq 0.0001$
R5	4	−3.62	0.31	0	0	$10 \geq 0.0001$
R6	1	−3.62	0.31	0	0	$10 \geq 0.0001$
R7	1	1	1	0	0	$0 = 1$
Kind	Std	Std	Std	Std	Std	Std
Type	Real	Real	Real	Int	Int	Int
Upper	Inf	Inf	Inf	1	1	1
Lower	0	0	0	0	0	0

variables, now is time to fill following N columns, corresponding to the K_i variables. There are as many K_i as the number of rows, and each K_i represents a block of $N-1$ constraints.

```
inf <- 1
sup <- nrows - 1
for(i in (ncols + 1):(ncols + nrows)){
  set.column(lpmo, i, replicate((nrows - 1), 10), indices=inf:sup)
  inf <- inf + nrows - 1
  sup <- sup + nrows - 1
}
```

Next, we set the objective function, constraint types, and right-hand-sides. The objective is the minimization of the K_i variables. These are binary variables, so the last N columns of the problem are defined as binary, the remainders are by default real values in the interval $[0, \infty)$. Finally, in the right-hand-side, we use ≥ 0.0001 because strict '$>$' cannot be used.

```
#Objective
set.objfn(lpmo, c(replicate(ncols, 0),replicate(nrows, 1))) #
    Constraint types

#Constraint types
set.type(lpmo, (ncols + 1):(ncols + nrows), "binary")

#right-hand-side
set.constr.type(lpmo, c(replicate((nrows * (nrows - 1)), ">="), "=
    "))
set.rhs(lpmo, c(replicate((nrows * (nrows - 1)), 0.0001), 1))
```

After this process, the *lpmo* object for this example should be the same as shown in Table 5. Now, to solve the proposed optimization problem we use the `solve()` function. That is,

```
solve(lpmo)

print(get.objective(lpmo))
print(get.variables(lpmo))
```

Table 6 Masked microdata: Microaggregated

V_1	V_2	V_3	V_4	V_5	V_6	V_7
13.50	33.00	121.58	35.33	51.33	802.00	34.25
13.50	33.00	121.58	35.33	51.33	2026.75	48.00
77.00	214.50	181.53	9.00	56.83	802.00	34.25
17.50	54.50	121.58	35.33	51.33	2026.75	48.00
22.00	69.50	181.53	9.00	56.83	802.00	34.25
51.50	76.00	181.53	9.00	56.83	481.50	15.00
17.50	54.50	181.53	9.00	56.83	481.50	15.00
36.50	166.00	121.58	35.33	51.33	2026.75	48.00
36.50	166.00	121.58	35.33	51.33	802.00	34.25
51.50	76.00	181.53	9.00	56.83	481.50	15.00
77.00	214.50	181.53	9.00	56.83	481.50	15.00
22.00	69.50	121.58	35.33	51.33	2026.75	48.00

The resolution of this toy example returns a value of 0 for the minimized objective function, it means that all the records have been correctly linked and the weighting vector returned is the optimal solution. That is, $[p_1 = 1, p_2 = 0, p_3 = 0]$, so only the first attribute, p_1, is necessary to correctly link all records.

We present another example of disclosure risk evaluation. In this example, for the sake of consistency we used the initial microdata example, Table 1, as original data. The protected file is a microaggregated version of this file. To appreciate the relevance of knowing the weights of each attribute, we apply different protection degrees to attributes. That is, we apply microaggregation to each pair of columns with a different parameter k (the higher the k, the higher the protection level). The *R* code is as follows,

```
#columns {1, 2} - protection degree k = 2
mic1 <- microaggregation(mdata[, 1:2], method="mdav", aggr=2)
#columns {3, 4, 5} - protection degree k = 6
mic2 <- microaggregation(mdata[, 3:5], method="mdav", aggr=6)
#columns {6, 7} - protection degree k = 4
mic3 <- microaggregation(mdata[, 6:7], method="mdav", aggr=4)

dMic <- as.table(cbind(mic1$mx, mic2$mx, mic3$mx))
```

Figure 6 shows how the protected data looks like after its anonymization taking into account three sets of columns and protecting each with a different protection level. Note that k-anonymity is not satisfied.

The solution of this problem leads to three records not correctly reidentified (i.e., 3 K_i different to zero) and to weights that give more importance to the variables with minimal perturbation (see Table 6, variables V_1 and V_2). So, there is a disclosure risk of 75 % (i.e., 9 over 12 records). Therefore, analyzing these weights from the protection entity point of view, more stringent protection measures have to be taken to these two first attributes.

Table 7 Improvement in the linkage ratio

Method	disclosure risk	p_1	p_2	p_3	p_4	p_5	p_6	p_7
AM	58.3%	0.143	0.143	0.143	0.143	0.143	0.143	0.143
d^2WM_p	75%	0.782	0.162	0	0	0	0.017	0.039

Table 7 compares this risk with the one we would have estimated only using an arithmetic mean (Euclidean distance or weighted distance with $p_i = 1/7$). That is, 58.3 %, which would underestimate the worse case scenario.

6 Summary

In this chapter some basic Privacy Preserving Data Mining techniques techniques were introduced and also the corresponding tools to perform them in *R*. We focused on generic or data-driven protection methods. i.e., they seek to modify data so that they can be published without giving away confidential information that can be linked to specific respondents and also achieve it with the minimum loss of information.

In overall we can conclude that *R* is a very good tool to carry out research on PPDM. Not only scientific research but also for prototyping. Engineers can easily test several protection methods and evaluate them, before development for production. Code from the examples of the chapter is available at https://github.com/dabril/r-book.

Acknowledgments Partial support by the Spanish MICINN (projects COPRIVACY (TIN2011-27076-C03-03), N-KHRONOUS (TIN2010-15764), and ARES (CONSOLIDER INGENIO 2010 CSD2007-00004)) and by the EC (FP7/2007-2013) Data without Boundaries (grant agreement number 262608) is acknowledged. The work contributed by the first author was carried out as part of the Computer Science Ph.D. program of the Universitat Autónoma de Barcelona (UAB).

References

1. Abril, D., Navarro-Arribas, G., Torra, V.: Supervised learning using mahalanobis distance for record linkage. In: Proceedings of 6th International Summer School on Aggregation Operators—AGOP2011. pp. 223–228 (2011)
2. Abril, D., Navarro-Arribas, G., Torra, V.: Improving record linkage with supervised learning for disclosure risk assessment. Inf. Fusion **13**(4), 274–284 (2012)
3. Abril, D., Navarro-Arribas, G., Torra, V.: Choquet integral for record linkage. Ann. Oper. Res. **195**, 97–110 (2012)
4. Abril, D., Navarro-Arribas, G., Torra, V.: Towards a private vector space model for confidential documents. In: Proceedings of the 28th Annual ACM Symposium on Applied Computing. pp. 944–945. SAC '13, ACM, New York, NY, USA (2013) http://doi.acm.org/10.1145/2480362.2480543
5. Agafitei, M., Defays, D.: Analysis of information loss in european data due to confidentiality. In: Joint UNECE/Eurostat work session on statistical data confidentiality (2011)

6. Agrawal, R., Srikant, R.: Privacy-preserving data mining. In: Proceedings of the ACM SIGMOD Conference on Management of Data. pp. 439–450. ACM Press (2000)
7. Brand, R.: Microdata protection through noise addition. In: Inference Control in Statistical Databases, from Theory to Practice. pp. 97–116. No. 2316 in Lecture Notes in Computer Science, Springer-Verlag (2002)
8. Defays, D., Nanopoulos, P.: Panels of enterprises and confidentiality: the small aggregates method. In: Proceedings of the 1992 Symposium on Design and Analysis of Longitudinal Surveys. pp. 195–204. Statistics Canada (1993)
9. Domingo-Ferrer, J., Mateo-Sanz, J.: Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans. Knowl. Data Eng. **14**, 189–201 (2002)
10. Domingo-Ferrer, J., Rebollo-Monedero, D.: Measuring risk and utility of anonymized data using information theory. In: Privacy and Anonymity in the Information Society (PAIS'09), Proceedings of the 2009 EDBT/ICDT Workshops (EDBT/ICDT '09). pp. 126–130. ACM (2009)
11. Domingo-Ferrer, J., Sebé, F., Castellà-Roca, J.: On the security of noise addition for privacy in statistical databases. In: Privacy in Statistical Databases. Lecture Notes In Computer Science, vol. 3050, pp. 149–161 (2004)
12. Domingo-Ferrer, J., Torra, V.: A quantitative comparison of disclosure control methods for microdata. In: Confidentiality, disclosure, and data access : theory and practical applications for statistical agencies, pp. 111–133. Elsevier (2001)
13. Domingo-Ferrer, J., Torra, V.: Ordinal, continous and heterogeneous anonymity through microaggregation. Data Min. Knowl. Disc. **11**(2), 195–212 (2005)
14. Hornik, K., Theussl, S.: Rglpk: R/GNU Linear Programming Kit Interface (2012), http://CRAN.R-project.org/package=Rglpk, R package version 0.3-8
15. Hubert, L., Arabie, P.: Comparing partitions. J. Classif. **2**(1), 193–218 (1985), http://dx.doi.org/10.1007/BF01908075
16. Jaro, M.A.: Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. J. Am. Stat. Assoc. **84**(406), 414–420 (1989)
17. lp_solve, Konis, K.: lpSolveAPI: R Interface for lp_solve version 5.5.2.0 (2011), http://CRAN.R-project.org/package=lpSolveAPI, R package version 5.5.2.0-5
18. Mateo-Sanz, J., Domingo-Ferrer, J., Sebé, F.: Probabilistic information loss measures in confidentiality protection of continuous microdata. Data Min. Knowl. Discov. **11**(2), 181–193 (2005)
19. Moore, R.: Controlled data swapping techniques for masking public use microdata sets. U.S. Bureau of the Census (unpublished manuscript) (1996)
20. Navarro-Arribas, G., Torra, V.: Privacy-preserving data-mining through microaggregation for web-based e-commerce. Internet Res. **20**(3), 366–384 (2010)
21. Navarro-Arribas, G., Torra, V., Erola, A., Castellà -Roca, J.: User k-anonymity for privacy preserving data mining of query logs. Inf. Process. Manage. **48**(3), 476–487 (2012)
22. Nin, J., Torra, V.: Towards the evaluation of time series protection methods. Inf. Sci. **179**(11), 1663–1677 (2009)
23. Oganian, A., Domingo-Ferrer, J.: On the complexity of optimal microaggregation for statistical disclosure control. Stat. J. United Nat. Econ. Comm. Eur. **18**, 345–354 (2001)
24. Pagliuca, D., Seri, G.: Some results of individual ranking method on the system of enterprise acounts annual survey. Esprit SDC Project, Delivrable MI-3/D2 (1999)
25. R Core Team: R data import/export (2012) http://cran.r-project.org/doc/manuals/R-data.pdf
26. Reiss, S.: Practical data-swapping: the first steps. In: IEEE Symposium on Security and Privacy. pp. 38–43 (1980)
27. Samarati, P.: Protecting respondents' identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)
28. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl. Based Syst. **10**(5), 557–570 (2002)
29. Sweeney, L.: Uniqueness of simple demographics in the U.S. population (2000)

30. Templ, M., Meindl, B.: Robust statistics meets sdc: New disclosure risk measures for continuous microdata masking. In: Proceedings of the UNESCO Chair in data privacy international conference on Privacy in Statistical Databases. pp. 177–189. Springer (2008)
31. Templ, M.: Statistical disclosure control for microdata using the r-package sdcmicro. Trans. Data Priv. **1**(2), 67–85 (2008)
32. Torra, V.: Microaggregation for categorical variables: a median based approach. In: Privacy in Statistical Databases. Lecture Notes in Computer Science, vol. 3050, pp. 162–174 (2004)
33. Torra, V.: Constrained microaggregation: adding constraints for data editing. Trans. Data Priv. **1**, 86–104 (2008)
34. Torra, V., Ladra, S.: Cluster-specific information loss measures in data privacy: A review. In: Third International Conference on Availability, Reliability and Security, 2008. ARES 08 (2008)
35. Torra, V., Navarro-Arribas, G.: Data privacy. WIREs Data Mining Knowl Discov (2014). doi:10.1002/widm.1129
36. Willenborg, L., de Waal, T.: Elements of Statistical Disclosure Control. Springer, Berliin (2001) (Lecture Notes in Statistics)

Optimisation-Based Study of Data Privacy by Using PRAM

Jordi Marés, Vicenç Torra and Natalie Shlomo

Abstract Dissemination of data with sensitive information has an implicit risk of unauthorised disclosure. Several masking methods have been developed in order to protect the data without losing too much information. One of the methods is called the Post Randomisation Method (PRAM) which is based on perturbations according to a Markov probability transition matrix. However, the method has the drawback that it is difficult to find an optimal transition matrix to perform perturbations which maximise data utility. In this paper we present an study of data privacy from the point of view of optimisation using evolutionary algorithms to generate optimal probability transition matrices. Optimality is with respect to a pre-defined fitness function which aims to preserve several data protection properties such as data utility and disclosure risk. We also provide experimental results using real datasets in order to illustrate and empirically evaluate the application of this technique.

1 Introduction

Data privacy became a very important issue since the first data publications in order to preserve the disclosure of sensitive information about individuals or institutions, but it has become even more important with the advances made in technology during

J. Marés (✉)
IIIA—Institut d'Investigació en Intelligència Artificial, CSIC—Consejo Superior de Investigaciones Científicas, Campus de Bellaterra, 08193 Bellaterra, Catalonia, Spain
e-mail: jmares@iiia.csic.es; jordimares@gmail.com

V. Torra
Institut d'Investigació en Intel·ligència Artificial, Consejo Superior de Investigaciones Científicas Campus de la UAB, 08193 Bellaterra, Catalonia, spain
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

N. Shlomo
Cathie Marsh Centre for Census and Survey Research (CCSR), The University of Manchester, Humanities Bridgeford Street, Manchester M13 9PL, UK
e-mail: natalie.shlomo@manchester.ac.uk

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_6

the last few decades. Nowadays the number of available datasets for statistical studies is growing more and more so the amount of sensitive data like the income and health illnesses is also growing. To avoid that a data release causes the disclosure of sensitive information statistical agencies and data owners need to be very careful. They must preserve the privacy of the people involved on this data.

The Statistical Disclosure Control (SDC) discipline is concerned with the anonymisation of the statistical data containing confidential information about individual entities such as individuals or institutions. SDC researchers have been working in the development of several data masking methods having as a final objective the construction of a masked dataset able to be released maintaining the privacy of the data respondents minimising the loss of information. However, masking methods are not enough. They also need a way to measure the performance of each masking method as well as the quality of the protection of a dataset. There exist two kind of measures to do this: the information loss and the disclosure risk [7].

Information loss measures check the quantity of harm inflected to the original data by the masking method, that is, it measures the amount of original information that has been lost during the masking process. There exist two different families of information loss measures: general measures and specific measures. On the other hand, general information loss measures roughly reflect the amount of information loss for a reasonable range of data uses. On the other hand, specific information loss measures evaluate the loss of statistical utility for a particular data analysis.

Disclosure risk evaluates the privacy of the respondents against possible malicious uses that third parties (sometimes called intruders) could do with the released information. Disclosure risk measures evaluate the number of respondents whose identity is revealed. Normally, these measures are computed in several scenarios where the intruder has partial knowledge of the original data. In order to compute the disclosure risk, general methods for re-identification are used. These methods find relationships (i.e. links) between the protected data and the partial knowledge which the intruder is assumed to have.

The problem here is that information loss and disclosure risk measures are inversely related. That is, if we perform an aggressive protection we will obtain a high information loss but a low disclosure risk. However, if we perform no protection (or a very light protection) we will obtain a low information loss and a high disclosure risk. Then, seeking a good protection that has low information loss and low disclosure risk is a difficult task. There exist many protection methods, each of them having different parameters to tweak. However, the protection process can be thought of as a function that takes the original data set and generates a new data set which should be optimised with respect to the concept of good protection (low information loss and low disclosure risk). Then, the protection process can be modelled as an optimisation problem and can be solved using state-of-the-art optimisation methods but, as the protection process is a very unknown and difficult task and the search space is large (the product of the domains size of all attributes to protect), this optimisation problem is not suitable to be solved using analytical methods. Evolutionary algorithms are a good choice in this case because they work well in this kind of situations.

In this paper we present a study of data privacy from the point of view of the optimisation. In order to do that we chose to work with the Post-Randomisation Method (PRAM) [9] as it was one of the most promising protection methods but it is not widely used because of the difficulty of finding a good parameterisation.

In the following sections we introduce different ways to use an evolutionary algorithm to get better protections. First, Sect. 2 presents the PRAM protection method. Section 3 provides a description about evolutionary algorithms. In Sect. 4 an evolutionary approach for PRAM matrices optimisation is presented. Section 5 introduces a way to use a genetic programming technique in the generation of matrices. Finally, Sect. 6 provides some concluding remarks.

2 The Post-Randomization Method

The Post Randomization method (PRAM) is a method for masking categorical variables in microdata files. In [20] the method and some of its implications were discussed in more detail. However, the PRAM method is still one of the least used for protecting microdata because of the difficulty in obtaining an optimal transition matrix to perform safe protections whilst maintaining data utility. This was demonstrated in the experiments carried out in [5] where the PRAM method was shown to have the worst utility and protection scores.

The PRAM method is as follows: Let t be the vector of frequencies and t/r the vector of relative frequencies of a categorical variable having L categories and r is the number of records in the microdata. Let P be a $L \times L$ probability transition matrix containing conditional probabilities: $p_{ij} = p(value_{perturbed} = j | value_{original} = i)$. In each record of the data, the category of the variable is changed or not changed according to the prescribed transition probabilities in the matrix P and the result of a draw of a random multinomial variate u with parameters p_{ij} $(j = 1, \ldots, L)$. If the j-th category is selected, category i is moved to category j. When $i = j$, no change occurs.

There are different ways to define the Markov matrices in the literature. We discuss here two of the approaches, which are the most commonly used. In the discussion we understand p_{kl} as the probability of changing a value k to a value l. Then, $\sum_{l=1}^{n} p_{kl} = 1$, where n is the number of categories. The first type is a fully-filled matrix with the off-diagonal elements depending on the corresponding frequencies in the original microdata file. This approach has been used in [2]. Formally, the probability p_{kl} for $k \neq l$ is defined by

$$p_{kl} = \frac{(1 - p_{kk}) \left(\sum_{i=1}^{n} \left(T_\xi(i) - T_\xi(k) - T_\xi(l) \right) \right)}{(n - 2) \left(\sum_{i=1}^{n} \left(T_\xi(i) - T_\xi(k) \right) \right)} \tag{1}$$

where $T_\xi(i)$ is the frequency of the category i inside the original dataset for the actual variable. In this approach p_{kk} is left as constant, that is, $p_{kk} = p$ for

all k. The key point of this equation is that it assigns the higher exchange probabilities to the categories with less frequency. In this way, the resultant dataset has more confusion.

The second type is a fully-filled matrix with the diagonal elements depending on the corresponding frequencies in the original microdata file. This approach has been used in [5]. In this case the row values are determined by the following expressions:

$$p_{kk} = 1 - \left(\theta T_\xi(K)/T_\xi(k)\right) \tag{2}$$

for $k = 1, \ldots, n$ and, then,

$$p_{kl} = \frac{1 - p_{kk}}{n - 1} \tag{3}$$

for $k \neq l$, where $T_\xi(K)$ is the smallest frequency greater than zero, and θ is a parameter in [0, 1].

3 Evolutionary Algorithms

Evolutionary algorithms are stochastic processes inspired by the model of biological evolution that was formulated for the first time by Charles Darwin, generally oriented to find exact or approximate solutions to optimisation or search problems [8, 11].

These algorithms maintain a population of individuals, denoted as $P(t)$ for generation t. Each individual $X'_j \in P(t)$ is evaluated by some measure of their "fitness". Fitness *evaluation* is used to guide individuals from generation to generation. Some of the selected members are *altered* by operators with an evolutive connotation, such as mutation and crossover. These operators create offspring from the existing population members from previous generations. Surviving individuals are evaluated again, and the process is repeated until some stopping criteria is reached.

Biological analogy was the original motivation for the genetic algorithm approach where in the selective breeding of plants or animals offspring are sought that have certain desirable characteristics which are determined at the genetic level by the way the parents' chromosomes combine. In the case of genetic algorithms, a population of strings is used, and these strings are often referred to in the genetic algorithms literature as *chromosomes*. The recombination of strings is carried out using simple analogies of genetic *crossover* and *mutation*, and the search is guided by the results of evaluating the objective function f (fitness function) for each string in the population. Based on this evaluation, strings that have higher *fitness* (i.e. represent better solutions) can be identified, and these are given more opportunity to breed.

An special case of evolutionary algorithms is the Genetic Programming (GP), which is an evolutionary computation technique that automatically solves problems without requiring the user to know or specify the form or structure of the solution in advance. At the most abstract level, GP is a systematic, domain-independent method

for getting computers to solve problems automatically starting from a high-level statement of what needs to be done.

GP stochastically transforms populations of programs into new, hopefully better, populations of programs. It is a random process so it can never guarantee results, like in the nature. However, this GP's essential randomness can lead it to escape traps which deterministic methods may be captured by and this has been very successful at evolving novel and unexpected ways of solving problems.

The creation of the initial random population is performed so as to create syntactically valid, executable programs. After the genetic operations are performed on the current generation of the population, the population of offspring replaces the old one. The tasks of measuring fitness, selection, and genetic operations are then iteratively repeated over many generations. The computer program resulting from this simulated process can be the solution to a given problem or a sequence of instructions for constructing the solution.

Evolutionary algorithms have already been used in other approaches to protect data like in [17] where the authors present a method for multivariate microaggregation based on genetic algorithms. Their results were very successful showing that this kind of algorithms are a good choice for these problems.

4 Evolutionary Optimisation of PRAM Matrices

As explained in Sect. 2, PRAM method is able to perform the same protections provided by many other state-of-the-art methods only by selecting the appropriate PRAM transition matrix. However, the difficulty of getting a good matrix to perform good protections is also the reason why this method is not widely used. Then, we can think that this matrix is the key point of the PRAM and it has to be the thing to be optimised.

The proposed method's algorithm is shown in Algorithm 1 and its main idea is to have an initial population P_0 of PRAM matrices (one per each attribute to protect) where each one is treated independently, this is, data is not changed across individuals (i.e. matrices), only inside the same individual. The reason for this is that matrices are of different sizes because it depends on the size of each attribute's domain and it would not be possible to properly exchange rows (or ranges of values) between a small matrix and a big one.

The different initial PRAM matrices X_{i_0} are being optimised through the iterations of the evolutionary algorithm where at generation t we produce a modified PRAM transition matrix represented by X_{i_t}. To produce the $X_{i_{t+1}}$ PRAM transition matrix at generation $t+1$, we generate an intermediate matrix X'_{i_t} resulting from applying a genetic operator to the current matrix X_{i_t}. The PRAM transition matrix $X_{i_{t+1}}$ at generation $t+1$ will be the one with better fitness (either X_{i_t} or X'_{i_t}) and the other discarded. This process is repeated at each generation.

Algorithm 1: Proposed Evolutionary Algorithm to Enhance PRAM Transition Matrices

Input: $P_0 = \{X_{0_0} ... X_{n_0}\}$, initial population of PRAM matrices
Output: $P'_t = \{X_{0_t} ... X_{n_t}\}$, optimised PRAM matrices after t generations
$t \Leftarrow 0$
$fitness\_eval(P_0)$
while $stopping(X_t) \neq true$; **do**
  $alter \Leftarrow$randomly choose between mutation and cross
  if $alter$ by mutation **then**
    $X'_{i_t} \Leftarrow mutate(X_{i_t})$
  else
    $X'_{i_t} \Leftarrow cross(X_{i_t})$
  end if
  if $fitness\_eval(X'_{i_t}) < fitness\_eval(X_{i_t})$ **then**
    $X_{i_{t+1}} \Leftarrow X'_{i_t}$
  else
    $X_{i_{t+1}} \Leftarrow X_{i_t}$
  end if
  $t \Leftarrow t + 1$
end while
return $\{X_{0_t} ... X_{n_t}\}$

Table 1 Example of values encoding

Decimal	Integer	Binary	Gray code
0.185	185	0010111001	0011100101

In the following sections we describe the key aspects of this algorithm such as the genotype encoding, the genetic operators, the fitness function, and the selection criteria.

4.1 Genotype Encoding

The initial probability transition matrices that we are trying to optimise contain probabilities with several decimals so in order to simplify it, all the probabilities are multiplied by 1,000 and only the integer part of the value is kept for the encoding an integer value.

After having the probabilities in integer format, the encoding of each individual X_i (i.e. the matrix corresponding to the ith attribute to protect) is done value by value transforming them into its Gray code representation. The choice of using Gray-coded representation was made to obtain fast and more accurate solutions than regular binary representations [10].

An encoding example is shown in Table 1. The example includes all the steps required during the entire encoding process.

4.2 Genetic Operators

The most common genetic operators in an evolutionary algorithm are mutation and crossover.

The main idea of mutation is to apply a slight random change in an individual of the population. In our case, the population consists only of a single individual (a Gray-coded matrix) so we will perform mutation by altering a single bit from a single Gray-coded number inside the transition matrix. To do that, both the bit and the number are chosen randomly.

The mutation used in this approach is performed as follows:

- Take a random value of the individual X and consider that the value at this position is x_i with $genome(x_i) = b_j b_{j-1}, \ldots, b_1$.
- Choose a bit position k at random, such that $1 \leq k \leq j$.
- Then a new individual is obtained by replacing the bit b_k by its negation counterpart, $b'_k = not(b_k)$.

In the crossover case, the general idea is to select two individuals from the population and generate two new individuals by concatenating a part of each one that is delimited by one or two crossing points chosen at random. In our approach, we wanted to not mix matrices between them as each one can have different size so we modified this operator to pick two ranges of values inside the individual, i.e. the PRAM transition matrix, delimited by two crossing points selected at random, and then swapping those two ranges inside the matrix to create a new matrix. The modified crossover could seem similar to a mutation but it is different in the sense that it produces big changes in the individual in order to test states far from the current one, while mutation produces small changes to test states closer to the current one.

Formally, we define the crossover of the individual X (a PRAM matrix) by swapping two ranges of values within the individual as follows:

- Take two value positions $\{s, r\}$ at random, and consider that the two values at this position are $x_s \in X$ and $x_r \in X$.
- Generate a random number m to indicate the length of the ranges. This number must be in the range $[0, \min(length(X) - s, length(X) - r, |s - r|)]$, where length$(X)$ is the total number of values inside the individual X, and $\|$ is the absolute value operator.
- Then the ranges $[x_s, x_{s+m}]$ and $[x_r, x_{r+m}]$ are swapped obtaining a new individual. For example, having $s < r$ and $X = \{x_1, \ldots, x_n\}$ the new individual will be $X' = \{x_1, \ldots, x_r, \ldots, x_{r+m}, \ldots, x_s, \ldots, x_{s+m}, \ldots, x_n\}$.

4.3 Fitness Function and Selection

The fitness function is the most important part of an evolutionary algorithm. It controls the convergence of the algorithm to the desired optimal solution, similar to

the objective function in mathematical programming. The main idea in our fitness function is to use the new probability transition matrix obtained after mutation and crossover to perturb the original data according to the PRAM method described in Sect. 2. The fitness function for the new set of transition matrices is calculated on the perturbed data and compared with the fitness of the perturbed data based on the current set of transition matrices.

The evaluation of PRAM matrices needs several steps before checking their protection quality. First of all, these PRAM matrices values are in Gray code representation so it is needed to restore them to floating point values. Then, it is not possible to check the quality of the matrices just by taking a look at them so, as a second step, we use these matrices to perform the multivariate PRAM protection on the original data obtaining a certain protected dataset. After this second step we are finally able to check the protection quality using the information loss and disclosure risk measures.

In the case of information loss measures we decided to use the average of the contingency table-based information loss (CTBIL) [5], distance-based information loss (DBIL) [5] and entropy-based information loss (EBIL) [12] (See Eq. 4). On the other hand, as disclosure risk measure we used the average of interval disclosure (ID) [4] and the maximum between distance-based record linkage (DBRL) [6] and probabilistic record linkage (PRL) [6] (See Eq. 5).

$$IL(X) = \frac{CTBIL(X) + DBIL(X) + EBIL(X)}{3} \tag{4}$$

$$DR(X) = \frac{ID(X) + \max(DBRL(X), PRL(X))}{2} \tag{5}$$

In this experiment we used two different kind of fitness functions because we performed executions based on different aspects. The first case is based on general purposes information loss and disclosure risk measures. It can be considered as a multi-objective optimisation problem because the goal is to minimise both measures. To solve this we used the Objective Weighting method giving the same importance to both Disclosure Risk (DR) and Information Loss (IL) measures, so both have $\frac{1}{2}$ as a weight value like in [15].

If F is the original file and $PRAM_{multivariate}(F, \{X_1, \ldots, X_n\})$ is the function that performs multivariate PRAM protection in F with the set of PRAM matrices $\{X'_1, \ldots, X'_n\}$, then, the score of the set of matrices $\{X_1, \ldots, X_n\}$ is computed as follows

$$\{X'_1, \ldots, X'_n\} = restore(\{X_1, \ldots, X_n\}) \tag{6}$$

$$F' = PRAM_{multivariate}(F, \{X'_1, \ldots, X'_n\}) \tag{7}$$

$$Score(\{X'_1, \ldots, X'_n\}) = \frac{DR(F') + IL(F')}{2} \tag{8}$$

where *DR*() is the disclosure risk evaluation function and *IL*() is the information loss evaluation function. Here the *restore* function is defined as the conversion from matrices of Gray-coded values to matrices of probability values.

Because the PRAM method takes random decisions in the protection step, the method can generate different protected files for the same Markov matrix, and they will also have different scores. In order to have more robust results, we compute five protected files for each candidate to be evaluated (i.e. each Markov matrix) and the average of their scores is taken as the candidate's final score. It should be noticed that the number of executions to perform is not fixed and it can be changed by the user. We used five executions to obtain more robust results without penalising too much the execution time. More formally:

$$FinalScore(\{X_1, \ldots, X_n\}) = \frac{\sum_{i=1}^{5} Score(\{X'_1, \ldots, X'_n\})}{5} \tag{9}$$

The second kind of fitness function has been used to test the information gain when adding the invariance property to the matrices. In this case to compute the fitness function of a certain PRAM transition matrix generated by the evolutionary algorithm we propose to use the difference in bivariate counts of two cross-classified categorical variables between the original data and the perturbed data where one of the categorical variables is perturbed with PRAM and the other categorical variable is not perturbed.

Formally, the fitness function is defined as follows

$$Fitness(R) = \frac{\sum_{ij} |counts_{original}(x_i, z_j) - counts_{perturbed}(x_i, z_j)|}{2 * \#records} \tag{10}$$

where x_i refers to the category i of attribute x, z_j refers to category j of attribute z, and || is the absolute value operator. It should be noted that only one of the attributes (x or z) is protected, the other one must be unprotected.

The use of this fitness function shows the optimisation of the transition matrix in preserving the frequency distribution of two cross-classified categorical variables in the perturbed data and whether it is similar to the distribution in the original data given that one of the categorical variables has been perturbed.

Before calculating the fitness function on each new generation of the transition matrix, we first need to carry out a pre-processing stage to ensure that the property of a probability transition matrix is fulfilled, i.e. each row of the matrix must add to one. This property can easily be lost when altering values with mutation and crossover. This is achieved by normalising the row by dividing each element by the sum of the entire row.

It should be noticed that some of the measures used in the fitness function are costly. However, as our approach is based on an evolutionary algorithm, it allows to easily use any other measures just by changing the fitness function (all the rest of the algorithm remains untouched). Thanks to this any improvement on these or other measures performance can be easily added to the approach.

4.4 Adding Invariance and Controlling Diagonal Values

A technique to boost the performance of probability transition matrices used for PRAM is to include the property of invariance [14]. This property ensures that the sufficient statistics of the protected attributes are preserved in expectation in the perturbed data and that the perturbed data is an unbiased moment estimator of the original data. In addition, controlling for the diagonal probabilities of the transition matrices ensures the desired level of perturbation according to the standards and thresholds set by data providers and also guarantees that the matrices can be inverted.

Placing the condition of invariance on the transition matrix P, i.e. $tP = t$ releases the users of the protected file of the extra effort to obtain an unbiased estimate of the original data, since t^* itself will be an unbiased estimate of t. The property of invariance means that the marginal distribution of the variable being perturbed is preserved in expectation.

In this work, the invariance is computed by following the two stage algorithm proposed in [19]. Let P be the PRAM matrix with $p_{jk} = p(c' = k|c = j)$ the probability of changing the value of category c equal to j to a new category c' equal to k. Now calculate the matrix Q using Bayes formula by $Q_{kj} = p(c = j|c' = k) = \frac{p_{jk}p(c=j)}{\sum_l p_{lk}p(c=l)}$. We estimate the entries of this matrix by $\frac{p_{jk}v_j}{\sum_l p_{lk}v_l}$, where v_j is the relative frequency of the category value j. For $R = PQ$ we obtain an invariant matrix where $vR = vPQ = v$ since $r_{ij} = \sum_k \frac{v_j p_{ik} p_{jk}}{\sum_l p_{lk} v_l}$ and $\sum_i v_i r_{ij} = \sum_k v_j p_{jk} = v_j$.

However, before making the transition matrix invariant, its diagonal dominance must be checked, that is, the diagonal probability of each row must be higher than the sum of all off-diagonal probabilities. That property is required to ensure that we are able to invert the transition matrix.

With respect to the property of diagonal dominance in the transition matrix, we also want to control the range of values that the diagonal of the matrix can have. This is because we do not want to have a very high probability of preserving the same category since then the related attribute will not be protected enough. On the other hand, we do not want to have a very low probability because it would mean the information contained on that attribute would be totally lost. For that reason we decided to force the diagonal values to be between 0.55 and 0.75 like in [14].

If a transition matrix contains a diagonal element below 0.55 we apply the approach shown in Eq. (11) to increase the value. On the other hand, if a matrix contains a diagonal element over 0.75 we use the approach shown in Eq. (13) to reduce the value. However, the diagonal dominance could be lost when reducing values. For that reason, after every execution of reducing values we test again for diagonal dominance.

Then, in order to ensure that a matrix is diagonal dominant we use the approach shown in [16] where the diagonal values are increased (and off diagonal are decreased proportionally) according to a parameter α. Equation (11) shows this approach where R' is the new PRAM matrix, I is the identity matrix and α is the control parameter. It should be noted that the higher the value of α, the smaller the increment in the diagonal values.

$$R' = \alpha R + (1 - \alpha) I \tag{11}$$

$$\beta = 0.75 / \max(p_{kk}) \tag{12}$$

$$p_{ij} = \begin{cases} \beta * p_{ij}, & \text{if i} = \text{j} \\ (\beta * p_{ij}) + \frac{1-\beta}{length(row_i)}, & \text{if i} \neq \text{j} \end{cases} \tag{13}$$

The invariance property is applied every time a new matrix is evaluated in the Fitness function.

4.5 Experimental Results

In this section we present the results of the experiments done to test the performance of our approach. These experiments were split in two parts. The first part shows the results regarding the general information loss and disclosure risk measures while in the second part shows the results regarding the addition of the invariance property to the PRAM matrices.

4.5.1 General Measures Testing Results

In order to illustrate and empirically evaluate our proposed method we used three different datasets to perform some experiments. The ones used in these experiments are the U.S. Housing Survey of 1993 from the U.S. Census Bureau [18], the German Credit [1] and the Solar Flare [1] datasets.

The U.S. Census Bureau dataset contains information about the size and the composition of the U.S. houses inventory on 1993. This dataset consists on 1,000 records represented in terms of 11 categorical attributes and 3 continuous attributes. The German Credit dataset describes German people's financial aspects. This dataset has 1,000 records with 7 numerical attributes and 13 categorical attributes. Finally, the Solar Flare dataset contains information about 1,389 different solar flares described with 10 categorical attributes.

For the U.S. Census dataset, Fig. 1 (left) shows the evolution of the Information Loss, Disclosure Risk and Score for this multi-attribute protections. During the evolutionary process it can be seen that there is a progressive decrement until around generation 1,100 of the Score value during all the process. Moreover, Disclosure Risk has suffered a big decrement, so the combination of this decrement with the little reduction of Information Loss has forced the Score value to be reduced.

Figure 1 (center) shows the evolution of the Information Loss, Disclosure Risk and Score for this multi-attribute protection in the case of the German Credit dataset. During the evolutionary process it can be seen that there is a quite progressive decrement of the Score value during all the process. Moreover the Disclosure Risk has

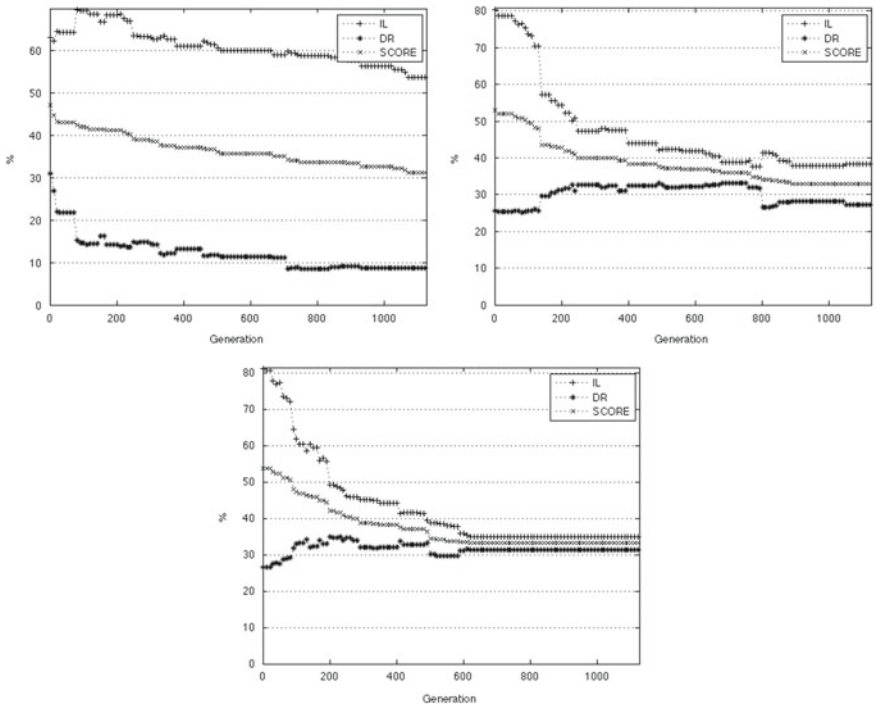


Fig. 1 Results for the protection of all three attributes at the same time in the U.S. Housing dataset (*left*), German Credit dataset (*center*) and Solar Flare dataset (*right*)

increased instead of decreased, but its final value is quite close to the initial one while the Information Loss has suffered a big decrement, so the combination of the two measures forced to reduce the Score value.

The results for the Solar Flare dataset are shown in Fig. 1 (right). In this figure it can be seen that all three measures have a fast stabilisation around generation 600. Disclosure risk has increased a little but Information Loss has been suffered a big decrement which causes an important reduction to the values of the Score measure.

In order to see the improvement provided by the evolutionary algorithm, Table 2 show the initial and final values for information loss, disclosure risk and score measures for all three datasets. It can be seen that in all cases the final score is much lower than the initial one, what means that the protection provided by the final matrix is better than the protection provided by the initial one. In addition, most of the cases, the values for information loss and disclosure risk are more balanced than the original ones what provides better trade-off between them.

Finally, in order to give an idea of the execution time by using the proposed fitness functions, in Table 3 we provide the results of the time spent per generation. It should be noticed that these times can vary significantly depending on when using different fitness functions.

Table 2 Initial and final scores for the protection of three attributes at the same time in each of the datasets

Dataset		IL	DR	Score
U.S. Housing	Initial	63.14	31.08	47.11
	Final	53.77	8.61	31.20
German Credit	Initial	80.36	25.63	52.99
	Final	38.39	27.30	32.84
Solar Flare	Initial	81.18	26.49	53.83
	Final	34.91	31.41	33.16

Table 3 Execution times for each type of operation in seconds

Operation	Average execution time
Mutation	120.34
Crossover	242.48

4.5.2 Invariance Information Gain Testing Results

In order to test the performance of our proposed approach, we show in this section analytical results obtained from a set of experiments made on the U.S. Housing Survey of 1993 dataset from the U.S. Census Bureau. We perturb the variable DEGREE and calculated the fitness function under two scenarios: crossing DEGREE with the unperturbed categorical variable SCH; and crossing DEGREE with the unperturbed categorical variable METRO. We chose these two scenarios because crossing DEGREE with the variable SCH represents the case where the bivariate counts distribution is skewed and crossing DEGREE with the variable METRO represents the case where the bivariate counts distribution is more uniform. This will test the performance of our approach under two extreme scenarios.

To evaluate the optimal probability transition matrix obtained from the evolutionary algorithm after 250 generations, we assess the data utility of the perturbed data using two measures: the difference in bivariate counts (which was also the measure used as the fitness function to guide the evolutionary algorithm) as shown in Eq. (10) and the relative absolute difference between the χ^2 test statistic calculated on the original and perturbed bivariate counts as shown in Eq. (14). The χ^2 statistic tests the association between categorical variables (see [3]).

$$\chi^2{}_{AbsDiff}(perturbed, original) = 100 * abs\left(\frac{\chi^2_{perturbed} - \chi^2_{original}}{\chi^2_{original}}\right) \tag{14}$$

The absolute relative difference in the χ^2 statistic provides an indication of attenuation of the association between the perturbed variable and the non-perturbed variable and whether we are moving towards the assumption of independence as a result of the perturbation.

Figure 2 shows the results of the percent difference in the bivariate counts as defined in Eq. (10) and the results of the relative absolute χ^2 difference between

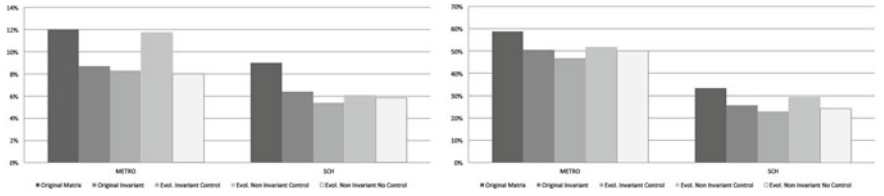


Fig. 2 Bivariates difference comparison (*left*) and the absolute relative difference in the χ^2 statistic comparison (*right*)

the original dataset and the dataset that was perturbed using the final probability transition matrix for the categorical variable DEGREE. Smaller percent differences show better results for data utility. The bivariate counts are produced by crossing the categorical variable DEGREE with the non-perturbed categorical variables METRO and SCH, respectively. It can be seen that in all cases the percent difference in bivariate counts has been reduced compared to using the original transition matrix to perturb DEGREE prior to the evolutionary algorithm.

In general, the transition matrix obtained by the evolutionary algorithm without the property of invariance does not perform as well as adding in the property of invariance. This is because we are dealing with a stochastic evolutionary algorithm and therefore it is more difficult to continually improve when we need to control the diagonal probabilities since the algorithm sometimes needs to go through a non-valid state, i.e. a matrix with diagonal values out of range, to reach a more optimal valid state afterwards.

Regarding the results relative absolute difference in the χ^2 statistic, the height of each bar represents the percent absolute difference between the χ^2 test statistic calculated on the original data compared to the perturbed data according to the same transition matrices evaluated above. The lower the bar means that there is less difference, i.e. the categories frequencies are more similar in both datasets. It can be seen that perturbing the variable DEGREE using the probability transition matrix obtained by the evolutionary algorithm with the property of invariance outperforms the other transition matrices, even in the case where the property of invariance is applied directly on the original transition matrix. There is more of a difference in the χ^2 test statistic based on the uniform bivariate counts of DEGREE crossed with METRO compared to the skewed bivariate counts of DEGREE crossed with SCH.

Based on the data utility measures, for the two cases of the evolutionary algorithm, we see that executing the algorithm with invariance provides, in general, probability transition matrices with better utility. In addition, our evolutionary algorithm is able to reach close results even without using the property of invariance. This is because of the use of the fitness function we have chosen. As we have used the differences between bivariate counts as the fitness function, this implies an indirect control over the marginal frequencies of the perturbed variable DEGREE, so ultimately we will have a similar effect to the case of applying the property of invariance. This fact proves that our evolutionary algorithm is able to learn this behaviour and to reach

Table 4 Summary of disclosure risk/data utility measures in the case of DEGREE-SCH

Matrix type	Data utility	Weighted average risk
Evol. invariant control	5.38	2.338
Evol. non-invariant control	6.10	2.722
Only invariance	6.41	2.827
Evol. non invariant no control	5.84	4.001
Initial matrix	8.99	1.780

transition matrices with similar effects compared to using the original transition matrix with the property of invariance.

To provide a disclosure risk-data utility summary and final assessment of the different probability transition matrices used to perturb the variable DEGREE in this experiment, Tables 4 and 5 show data utility and disclosure risk measures. The data utility measure is the difference in bivariate counts as shown in Fig. 2. We use the disclosure risk measure described in [9] which is defined for each value (attribute) of the categorical variable. This measure computes the ratio of the expected number of records in the perturbed file with a value k equal to its value in the original dataset divided by the expected number of records in the perturbed file with a value k that arises from a different value in the original dataset. Hence, the smaller the value of the expectation ratio, the more likely it is that a record in the perturbed file with value k did not originally belong to this category, and thus the more protection in the perturbed file. To obtain the single disclosure risk measure, we calculated a weighted average of the disclosure risk measures by taking into account the frequency of each possible value of DEGREE.

For the case of the skewed bivariate counts of DEGREE crossed with SCH in Table 4, we can see that the transition matrix with the highest data utility (selected with the smallest percent difference in the bivariate counts) is the matrix generated by the evolutionary algorithm with the property of invariance. This increased the disclosure risk from 1.780 based on the original transition matrix to 2.338, meaning that there are more values on average in the perturbed data that were not changed. The original transition matrix with the property of invariance had higher disclosure risk and lower data utility compared to the matrix resulting from the evolutionary algorithm with the property of invariance. In addition, the matrix obtained with the

Table 5 Summary of disclosure risk/data utility measures in the case of DEGREE-METRO

Matrix type	Data utility	Weighted average risk
Evol. invariant control	8.27	2.826
Only invariance	8.69	2.827
Evol. non invariant no control	8.01	4.316
Evol. non-invariant control	11.74	2.549
Initial matrix	12.00	1.780

evolutionary approach without the property of invariance nor the diagonal values control is the one with the worst results (before the initial matrix).

The case of uniform bivariate counts when DEGREE is crossed with METRO is shown in Table 5, we again see that the transition matrix with the highest data utility is the matrix generated by the evolutionary algorithm with the property of invariance, but at higher disclosure risk compared to the bivariate counts of DEGREE and SCH. From Table 5, the original matrix with the property of invariance and the evolutionary algorithm with the property of invariance had similar results with respect to disclosure risk and data utility. In this case, using the matrix obtained with the evolutionary approach without the property of invariance nor the diagonal values control, we obtained a similar behaviour than in the skewed bivariate counts showing that applying invariance property boosts the performance of the evolutionary approach.

5 Genetic Programming Optimisation Approach

In this section we present an approach based on genetic programming to go one step further in the optimisation of the PRAM matrices. In this case, instead of dealing with the values inside the matrix itself, we want to optimise the analytical function that generates the matrix. The outline of this approach is shown in Algorithm 2, which is based in the genetic programming's steady-state algorithm, which works by randomly selecting some individuals from the population and then keeping the best of the selected for the next generation's population and replacing the worst by the generated offspring under a certain replacement conditions. This approach ensures that we are not going to loose a good solution from the population.

The algorithm maintains a population of $pop_{\max}$ individuals where each one is an equation to build a PRAM matrix. Then, in each generation, a random subset of individuals is selected from the population and the selected programs are evaluated. Using these fitness results, selected individuals are split between winners and loosers based on their fitness. Winners are going to be crossed or mutated in order to try to further improve their fitness, and loosers will might be substituted in the population for the next generation. Finally, the fitness of the new offspring is computed, followed by a checking of whether they are going to replace the loosers in the next generation. When the algorithm finishes, the best individual in the population in terms of fitness value is returned.

In the following sections we describe how to represent and initialise the population and how to compute the fitness of the individuals.

5.1 Population Representation and Initialisation

In the problem of finding analytical PRAM matrices we have to deal with equations that are going to be used to build the transition matrices. Then, in our genetic programming approach we have to initialise a population with equations. These equations

Algorithm 2: GP Steady-State Algorithm for Seeking PRAM Matrices Analytically.

Input: X original dataset, pop_{max} maximum number of programs in the population, gen_{max} maximum number of generatons, $cross_{rate}$ crossover rate
Output: $best_p$ best PRAM matrix equation in final population
$P \Leftarrow$ initializePopulation(pop_{max})
$t \Leftarrow 0$
while $t < gen_{max}$ **do**
  $S \Leftarrow$ selectSubset(P_t,sel_{max})
  $F_S \Leftarrow$ computeFitness(S,X)
  $[Winners, Loosers] \Leftarrow$ selectWinners(F_S,S)
  $a \Leftarrow$ randomNumberBetween(0,1)
  if $a < cross_{rate}$ **then**
    $newInds \Leftarrow$ cross($Winners$)
  else
    $newInds \Leftarrow$ mutate($Winners$)
  end if
  $F_{newInds} \Leftarrow$ computeFitness($newInds$,X)
  $P_{t+1} \Leftarrow$ replace($Winners$,$Loosers$,P_t)
  $t \Leftarrow t + 1$
end while
$best_p \Leftarrow$ selectBestProgram(P_t)
return $best_p$

are represented as tree structures by using terminal nodes in the leafs and function nodes in the internals.

In our case we defined the terminals set as $T = \{N, freq_{\max}, freq_{\min}, freq_i, freq_j\}$. N represents the total number of records in the dataset, $freq_{\max}$ is the maximum from all catgories frequencies, $freq_{\min}$ is the minimum of all categories frequencies, $freq_i$ is the frequency of the ith original category, $freq_j$ is the frequency of the jth category that the ith category can be changed to.

Regarding the functions set we decided to define it as $F = \{sum, sub, mul, div\}$ representing the summation, the subtraction, the multiplication and the division arithmetic functions. This selection of functions was driven by the fact that having a function set too large can make the search for a solution harder and that to have a good function set it should only include arithmetic and logic operators. In our case, logic operators do not make sense, so we decided to use only the basic arithmetic operators.

Once we had defined the terminals and functions sets, the population was built using the half-and-half approach to ensure the diversity of individuals. This method works by dividing the population amongst individuals to be initialised with trees having depths $1, \ldots, depth_{\max}$ [13], where the $depth_{\max}$ is set by the user.

5.2 Mutation and Crossover Operators

There exist several ways to perform mutation on tree structures. In this case we decided to use the subtree mutation method, which is the most widely used in GP and it works by randomly selecting a node in a tree and it is changed by a new random

subtree. It should be noticed that the newly created random subtree must fulfil the restriction of not exceeding the maximum individual depth when it is added to the mutated tree.

The election was based on the freedom this method gives to create new shapes for the mutated trees, this is, mutating a tree at a node n_i with a certain level l_{n_i} can produce a new subgraph with any level in the range $[1, level_{\max} - l_{n_i}]$ and each branch can have different length as well. Therefore, by applying subtree mutation we allow the genetic programming algorithm be more creative when altering known solutions to find new and better ones.

In the case of the crossover we used a tree crossover approach, which works by swapping two randomly selected subtrees between two individuals by doing the following steps.

- Select a random node from each of the two equation trees to cross.
- Determine in each tree the maximal subtrees that have the selected nodes as root.
- Swap the two subtrees between the two trees.

To do that, we only added the following constraints based on the selected nodes levels

$$level_{n1_i} + subtreeDepth_{n2_j} \leq level_{\max}$$

$$level_{n2_i} + subtreeDepth_{n1_j} \leq level_{\max}$$

where $level_{n2_i}$ is the level of the selected node in the second tree, $level_{n1_j}$ is the level of the selected node in the first tree, $subtreeDepth_x$ is the maximum depth of the subtree starting in node x and $level_{\max}$ is the maximum number of levels allowed in a tree.

5.3 Fitness and Replacement

In order to guide the improvement of the programs in the population we have to define a fitness function to be applied on them. In this case, this function has several steps to follow:

1. The program (equation) to evaluate has to be executed to build a real PRAM matrix.
2. The obtained PRAM matrix has to be used to protect the original dataset.
3. Information loss and disclosure risk measures have to be calculated for the masked dataset.
4. The fitness value for the given equation will be the maximum value between both mesures.

In order to execute the programs we have to transform the tree structures into a real executable programs. In our case, we have to add the tools to be able to

execute the equations to get their associated PRAM matrix. However, this is an easy task because our tree-based representation allows us to obtain a computation of the represented equation just by traversing the tree in post-orden. By doing that we have that any function will be preceded in the final list by its operands. Then, we simply go through this list of operands and operations saving terminal values in a stack and when a function is found we take the two first elements from the stack to use them in the function and the results is saved again in the stack. At the end, we end up with only one element in the stack which is the final result of the execution and it is returned.

Next important point from the four steps to follow in the fitness function shown above is the election of information loss and disclosure risk measures to be used when evaluating the protected dataset. This point is a key one because these measures are the ones that will guide our approach to evolve towards a solution with better and minimised trade-off between them.

In this approach we used the same generic information loss and disclosure risk measures than in the approach shown in the previous section. To aggregate these measures we used two different approaches to compare their behaviours. The first one is to take the maximum of the two values as shown in Eq. (15), and the second one is to take the average of both values as shown in Eq. (16).

$$Score(X) = \max(IL(X), DR(X)) \tag{15}$$

$$Score(X) = \frac{IL(X) + DR(X)}{2} \tag{16}$$

At the end of the evaluation, we have to decide whether the new offspring will be part of the population for the next generation. In our case, the replacement is done by comparing the fitness score of the new offspring and the fitness of the tournament loosers in the selection process. We keep taking the program inside the loosers set which has the lowest fitness score and we replace it with the new offspring with the highest score. This process continues with the other programs until we have checked all new offspring programs.

5.4 Experimental Results

In this section we present the experimental results to show the performance of our approach. To do that we used the U.S. Housing, German Credit and Solar Flare datasets introduced in Sect. 4. In this case we performed a multivariate protection of three attributes in each dataset. For the U.S. Housing dataset we protected the BUILT, DEGREE and GRADE1 attributes. For the German Credit dataset we protected the EXISTACC, SAVINGS and PRESEMPLOY attributes. Finally, for the Solar Flare dataset we protected the CLASS, LARGSPOT and SPOTDIST attributes.

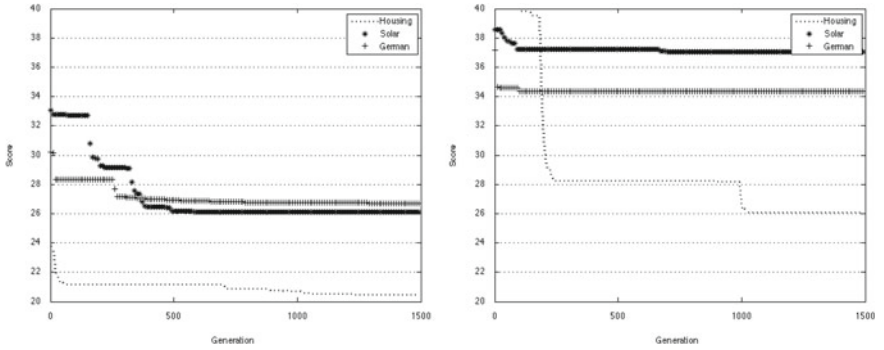


Fig. 3 Best equation's scores evolution using the mean (*left*) and max (*right*) fitness function

The experiments consisted on running 1,500 generations of the genetic programming approach for each dataset several times with different configurations in order to compare them. These configurations were the different combinations of maximum depth of the tree-based equations in the population and the two fitness measures we wanted to test. Regarding the maximum depth of the tree-based equations we used trees of five levels at maximum in order to be able to test simple and also complex generated equations. To have this variety of levels in the population we initialized it using the half-and-half approach with a population size of six equations.

Figure 3 shows the evolution of the best solution's score during the 1,500 generations in all three datasets using the two different (max and mean) proposed fitness functions. It can be seen that in all datasets we achieved an improvement of the best solution's score.

In the case of using the mean fitness function we can see that we got a significant improvement for each of the datasets. The U.S. Housing dataset's best solution went from a score of 23.77 to a score of 20.45, the German Credit dataset's one went from 30.23 to 26.72 and the Solar Flare's one from 33.09 to 26.10.

On the other hand, if we take a look at the results from the executions using the max fitness function we can see that we had been able to improve all datasets protections again. The U.S. Housing dataset's best solution went from a score of 39.96 to a score of 26.03, the German Credit dataset's one went from 37.18 to 34.33 and the Solar Flare's one from 38.57 to 37.03. However, in the cases of German Credit and Solar Flare datasets, we experimented much less improvement than in the U.S. Housing one. The reason for that is that this second fitness function is much more strict than the first one as it will only improve if the maximum value between IL and DR is decreased, while in the mean fitness function case it will improve when any change of their values makes the average decrease (for example, if we have $IL = 20$ and $DR = 30$, this function will think the individual improves if it goes to $IL = 5$, $DR = 40$). This behaviour then makes it more difficult to improve the datasets with a small number of available categories per attribute like these two because changing a category in these datasets causes more abrupt impact on IL and DR.

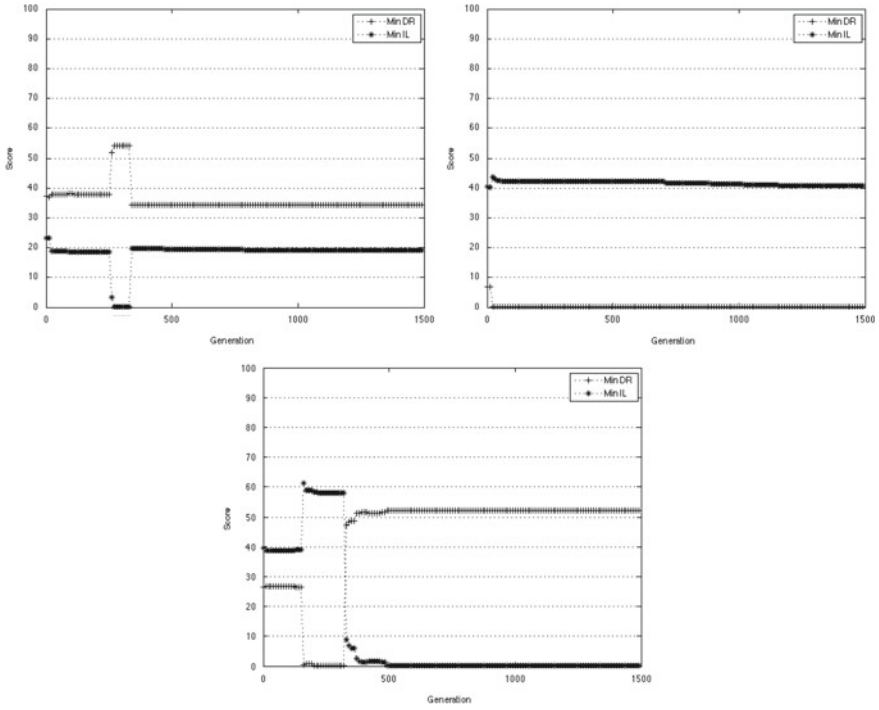


Fig. 4 Best equation's IL and DR evolution for the German Credit dataset (*left*), the U.S. Housing dataset (*center*) and the Solar Flare dataset (*right*) using the mean fitness function

It may seem as if after seeing the results of the score evolution, the mean fitness function is the one that performs better protections. Figures 4 and 5 show the evolution of the same executions presented above but decomposing the score with the information loss and disclosure risk evolutions. There, it can be seen that in all cases when using the mean fitness function we obtained a very irregular behaviour with very distant values of information loss and disclosure risk. However, when using the max fitness function we got a very different behaviour having a more controlled evolution and a kind of converging behaviour of the two measures. Then, taking into account that we want to achieve protections with low and balanced values for both measures, we can say that the max fitness function performs better protections than the mean fitness function.

This fact of having better protections using the max fitness function can be seen more detailed in Tables 6, 7 and 8 which show, for each dataset, the best protections using the two most used state-of-the-art equations to build PRAM matrices, and our genetic programming approach with the two different fitness functions.

In the case of the German Credit dataset (Table 6) it can be seen that using the uniform PRAM matrix results in a very bad protected dataset with very unbalanced information loss and disclosure risk measures and that using the frequency-based

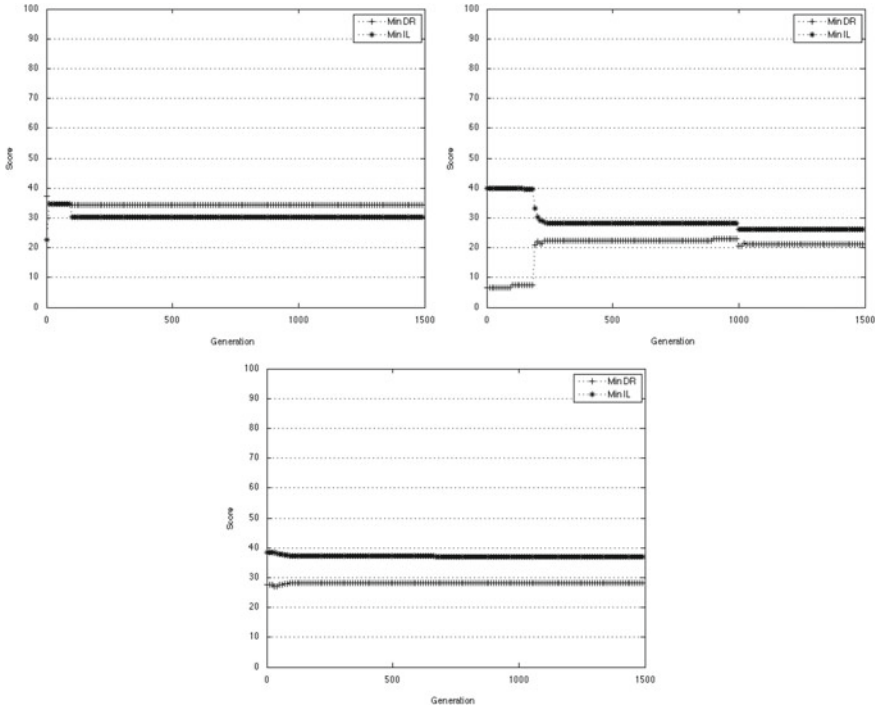


Fig. 5 Best equation's IL and DR evolution for the German Credit dataset (*left*), the U.S. Housing dataset (*center*) and the Solar Flare dataset (*right*) using the max fitness function

Table 6 Measures comparison between standard PRAM and the output of our approach for the German Credit dataset

Protection	*IL*	*DR*	$SCORE_{\max}$	$SCORE_{mean}$
Freq PRAM	37.88	31.03	37.88	34.45
Unif PRAM	13.60	46.12	46.12	29.86
GP max	30.39	34.33	34.33	32.36
GP mean	19.11	34.33	34.33	26.72

Table 7 Measures comparison between standard PRAM and the output of our approach for the Solar Flare dataset

Protection	*IL*	*DR*	$SCORE_{\max}$	$SCORE_{mean}$
Freq PRAM	29.26	36.09	36.09	32.67
Unif PRAM	16.06	42.68	42.68	29.38
GP max	37.03	28.25	37.03	32.64
GP mean	0.03	52.19	52.19	26.10

Table 8 Measures comparison between standard PRAM and the output of our approach for the U.S. Housing dataset

Protection	*IL*	*DR*	$SCORE_{\max}$	$SCORE_{mean}$
Freq PRAM	32.31	29.18	32.31	30.74
Unif PRAM	11.92	54.15	54.15	33.04
GP max	26.03	21.10	26.03	23.57
GP mean	40.84	0.09	40.84	20.45

PRAM matrix results in a quite balanced measures values. However, our genetic programming approaches using max and mean fitness functions outperformed the state-of-the-art measures. That is, we obtained better balanced and lower values in the genetic programming using max fitness function than in the frequency-based state-of-the-art matrix case and we also obtained more balanced and better trade-off using the genetic programming with mean fitness function than using the state-of-the-art uniform PRAM matrix.

Table 7 shows the results of the Solar Flare dataset and we can observe a similar behaviour as in the previous dataset. However, in this case we obtained very bad results in the case of using the mean fitness function in our genetic programming approach because it generated a protection with a very unbalanced measures values. This fact shows that, for this dataset, the mean fitness function is not useful.

The results of the U.S. Housing dataset are shown in Table 8. For this last case, the results follow the same pattern. We obtained a significant improvement using the genetic programming approach with the max fitness function, and the results show again that using the mean function is a bad idea because it leads again to protections with bad trade-off between measures.

To wrap it up, Fig. 6 shows, for each dataset, the position of the best protections for each approach in the space of values for information loss and disclosure risk. As said before, our goal is to obtain good protections and those protections will be the ones with balanced and low pair of values for information loss and disclosure risk. In these scatter plots, a good protection will be placed close to the diagonal and close to the ideal point (but impossible) (0, 0) where we would not have any information loss and no disclosure risk. It can be seen that in all cases the protections made by our genetic programming approach using the max fitness function are located in this area of good protections and it also beats the best state-of-the-art matrix: the frequency-based PRAM matrix. Here it can be seen again that the protections generated by the genetic programming using the mean fitness function are bad as they fall too far away from the good protections region.

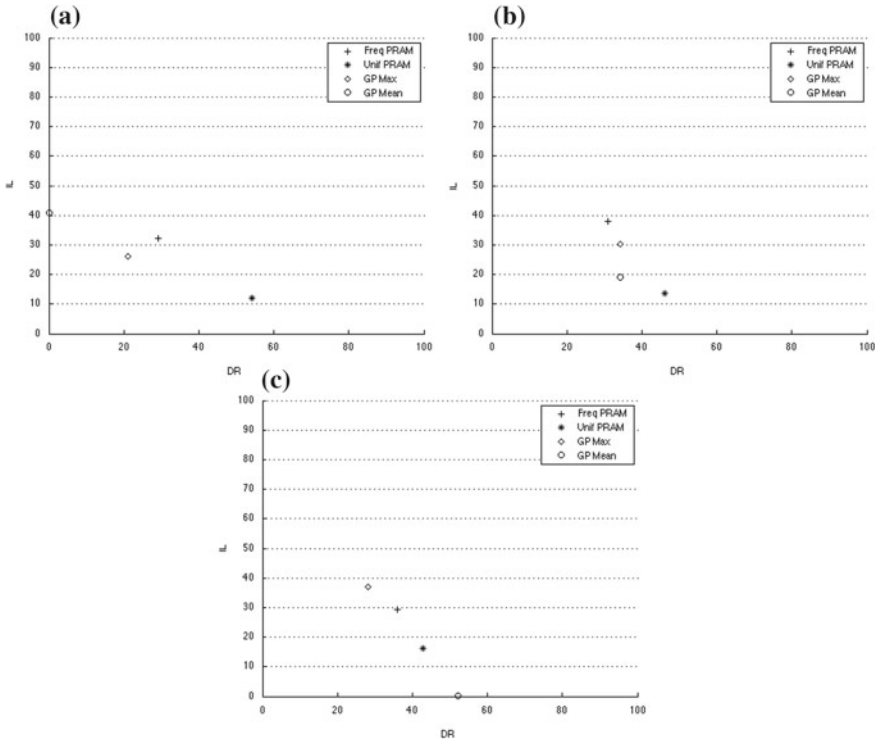


Fig. 6 Scatter plot with the best protections for each PRAM method. **a** U.S. Housing dataset. **b** German Credit dataset. **c** Solar Flare dataset

6 Conclusions

In this work we have presented an study of how to tackle the problem of finding a more effective PRAM matrix by facing it as an optimisation problem. In order to do that, we proposed the usage of evolutionary algorithms, which guide the solutions in the algorithm's population towards the optimum one respect to the provided fitness function. In addition we have seen that it is possible to use evolutionary algorithms to optimise transition matrices for the PRAM protection method obtaining matrices that perform better protections. It is difficult to obtain good matrices analytically so using the evolutionary algorithm it makes the task easier for us. Finally, we have shown how to add additional properties to these PRAM matrices like invariance. This property makes the evolutionary algorithm produce matrices that perform protections with better data utility.

We have also shown a way to optimise the PRAM matrices by embedding the analytical equations used to create these matrices in a genetic programming algorithm. Although there is much work still to be done in this aspect, it can be concluded that genetic programming can be a good approach to find new and enhanced PRAM

matrix equations. We compared two different aggregation functions to compute the fitness (max and mean functions) and the best one has been the max function as it generated equations that performed protections with much better balance between information loss and disclosure risk, and with lower values in these measures. It has also been proven that, in almost all cases, our genetic programming approach has beated the performance of the two most used state-of-the-art PRAM matrix equations.

Acknowledgments This work has been partially supported by the Spanish MECARES-CONSOLIDER INGENIO 2010 CSD2007-00004, and COPRIVACY TIN2011-27076-C03-03 and the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement num. 262608.

References

1. Bache, K., Lichman, M.: UCI machine learning repository. http://archive.ics.uci.edu/ml (2013)
2. De Wolf, P., Van Gelder, I.: An empirical evaluation of PRAM. Discussion Paper No. 04012. Statistics Netherlands, Voorburg/Heerlen (2004)
3. DeGroot, M., Schervish, M.: Probability and Statistics. Addison-Wesley Series in Statistics, 4th edn. Addison-Wesley, Boston (2012)
4. Domingo-Ferrer, J., Torra, V.: Disclosure control methods and information loss for microdata. In: Doyle, P., Lane, J.I., Theuwes, J.J.M., Vatz, L. (eds.) Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 91–110. Elsevier, Amsterdam (2001) (chap. 5)
5. Domingo-Ferrer, J., Torra, V.: A quantitative comparison of disclosure control methods for microdata. In: Doyle, P., Lane, J.I., Theuwes, J.J.M., Zayatz, L. (eds.) Confidentiality, isclosure, and Data Access : Theory and Practical Applications for Statistical Agencies, pp. 111–133. Elsevier, Amsterdam (2001)
6. Domingo-Ferrer, J., Torra, V.: Distance-based and probabilistic record linkage for re-identification of records with categorical variables. Butlletí de lACIA **28**, 243–250 (2002)
7. Fienberg, S.: Conflict between the needs for access to statistical information and demands for confidentiality. J. Off. Stat. **10**(2), 115–132 (1994)
8. Goldberg, D.: Genetic Algorithms in Search, Optimization and Machine Learning, 1st edn. Addison-Wesley Longman Publishing Co. Inc., Boston (1989)
9. Gouweleeuw, J., Kooiman, P., Willenborg, L., de Wolf, P.: Post randomization for statistical disclosure control: theory and implementation. J. Off. Stat. **14**(4), 463–478 (1998)
10. Greiner, D., Winter, G., Emperador, J.M., Galván, B.: Gray coding in evolutionary multicriteria optimization: application in frame structural optimum design. In: Proceedings of the Third international conference on Evolutionary Multi-Criterion Optimization, pp. 576–591. EMO'05, Springer, Berlin, Heidelberg (2005). http://dx.doi.org/10.1007/978-3-540-31880-4_40
11. Holland, J.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975) (2nd edn.: MIT Press, 1992)
12. Kooiman, P., Willenborg, L., Gouweleeuw, J.: PRAM: a method for disclosure limitation of microdata. Research Paper No. 9705. Statistics Netherlands, Voorburg, (1997)
13. Koza, J.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge (1992)
14. Marés, J., Shlomo, N.: Data privacy using an evolutionary algorithm for invariant PRAM matrices. Comput. Stat. Data Anal. **79**, 1–13 (2014)

15. Mars, J., Torra, V.: An evolutionary algorithm to enhance multivariate post-randomization method (PRAM) protections. Inf. Sci. (0) (2014). http://www.sciencedirect.com/science/article/pii/S002002551400348X
16. Shlomo, N., Young, C.: Invariant post-tabular protection of census frequency counts. In: Domingo-Ferrer, J., Saygin, Y. (eds.) Privacy in Statistical Databases. Lecture Notes in Computer Science, vol. 5262, pp. 77–89. Springer, Berlin (2008). http://dblp.uni-trier.de/db/conf/psd/psd2008.html#ShlomoY08
17. Solanas, A., Martinez-Balleste, A., Mateo-Sanz, J., Domingo-Ferrer, J.: Multivariate microaggregation based genetic algorithms. In: 3rd International IEEE Conference on Intelligent Systems 2006, pp. 65–70, Sept 2006
18. U.S. Census Bureau: U.S. Housing Survey of 1993 (1993), http://quickfacts.census.gov
19. Willenborg, L., Waal, T.D.: Elements of Statistical Disclosure Control. In: Lecture Notes in Statistics, vol. 155. Springer, Berlin (2000)
20. Wolf, P.D., Gouweleeuw, J., Kooiman, P., Willenborg, L.: Reflections on PRAM. In: Statistical Data Protection, pp. 337–349. Office for Official Publications of the European Communities, Luxembourg (1998)

Part III
Respondent Privacy: Semantic Related Respondent Privacy Protection

Semantic Anonymisation of Categorical Datasets

Sergio Martínez, Aida Valls and David Sánchez

Abstract The exploitation of microdata compiled by statistical agencies is of great interest for the data mining community. However, such data often include sensitive information that can be directly or indirectly related to individuals. Hence, an appropriate anonymisation process is needed to minimise the risk of disclosing identities and/or confidential data. In the past, many anonymisation methods have been developed to deal with numerical data, but approaches tackling the anonymisation of non-numerical values (e.g. categorical, textual) are scarce and shallow. Since the utility of this kind of information is closely related to the preservation of its meaning, in this work, the notion of semantic similarity is used to enable a semantically coherent anonymisation. The knowledge modelled in ontologies is used as the basic pillar to propose semantic operators that enable an accurate management and transformation of categorical attributes. These operators are then used in three anonymisation mechanisms: Semantic Recoding, Semantic and Adaptive Microaggregation and Semantic Resampling. The three algorithms are compared in terms of semantic utility, privacy disclosure risk and runtime, with encouraging results.

1 Introduction

Inference control in statistical databases or Statistical Disclosure Control (SDC) aims to disseminate statistical data while preserving confidentiality. SDC is focused mainly on the preservation of privacy in structured databases [1–6]. Statistical Disclosure Control methods transform the original database into a new database by means on an anonymisation procedure, which takes into account that the protected data satisfies

S. Martínez (✉) · A. Valls · D. Sánchez
Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili,
Avda. Països Catalans, 26, 43007 Tarragona, Catalonia, Spain
e-mail: sergio.martinezl@urv.cat

A. Valls
e-mail: aida.valls@urv.cat

D. Sánchez
e-mail: david.sanchez@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_7

simultaneously utility and security conditions. The dataset will be useful if it is representative of the original dataset and it will be secure if it doesnot allow the re-identification of the original data.

In this chapter we focus only on the protection of *microdata* files, which consist on records detailing the information of an individual (e.g. person or company). The release of midrodata implies a risk of disclosure of the information of that individual. Given an original microdata set *D*with *n* records (corresponding *n* individuals) and *m* values in each record (corresponding to *m* attributes that are not identifiers), the goal is to release a protected microdata set D^A (with also *n* records an *m* attributes) in such a way that:

1. Disclosure risk (i.e. the risk that a user or an intruder can use D^A to determine confidential attributes on a specific individual among those in *D*) is low.
2. Data analysis (regressions, means, data mining, etc.) on D^A and on *D* yield the same or at least similar results.

Most privacy-preserving mechanisms classify attributes in a dataset as: *Identifiers* (attributes that unambiguously identify the individual), *Quasi-identifiers* (attributes that may identify some of the respondents if they are combined with the other attributes available in external sources) and *Confidential*(attributes that contain sensitive information).

A well-known privacy model that relies on such classification to offer an a priori privacy guarantee is the *k*-anonymity [7]. *A dataset is said to satisfy k-anonymity for* $k > 1$ *if, for each combination of values of quasi-identifier attributes (e.g. name, address, age, gender, etc.), at least k records exist in the dataset sharing that combination.* Once a value for *k* is fixed (which should be a value that keeps the re-identification risk low enough), the goal of the masking method is to find an anonymisation that minimises the information loss (i.e. maximises the quality of the dataset D^A).

Anonymisation mechanisms enforcing *k*-anonymity convert an original set *D* in a publishable set D^A through a masking process. These methods distort quasi-identifiers in a way that unique combinations of values in the original dataset disappear and data becomes more homogenous.

Notice, that the posterior use of the data plays an important role in the anonymisation process because the masked version should enable extracting the same conclusions from data analysis than the original one. This is especially important with data mining analysis, such as clustering, rule induction, profiling, or prediction, among others. In fact, privacy preserving data mining is a new research field that attempts to develop tools to study in an integrated way how to deal with privacy issues while performing data analysis [8]. The quality of the data can be measured as a function of the distribution of the values in the datasets. Even though data distribution is a dimension of data utility, we argue, as it has been stated by other authors [9], that retaining the semantics of the dataset with non-numerical data plays a more important role when one aims to extract conclusions by means of data analysis techniques.

Most masking method enforcing *k*-anonymity focus on the protection of numerical attributes. However, applying these methods to non-numerical attributes is not

straightforward because of the limitations on defining appropriate operators to manage categorical values. Although some recent masking methods are able to manage categorical data, they do not make an appropriate use of semantic techniques to preserve the *meaning* of the original dataset.

Methods available in the literature which consider the semantics of categorical attributes are mainly based on data *recoding*. To do so, they generalise input values by relying on tailor-made hierarchical structures: Value Generalization Hierarchies (VGHs). This approach presents three main drawbacks:

1. VGHs are manually constructed from each attribute in function of the input data (categorical values correspond to leaves). Human intervention is required and the VGH is only valid for a concrete dataset. This fact may not be assumable when dealing with large sets of categories.
2. VGHs produce ad-hoc and small hierarchies with a much reduced taxonomical detail offering a rough and biased knowledge model.
3. Generalisations based on VGHs usually produce a high information loss due to their coarse granularity. Moreover, the quality of the results heavily depends on the structure of VGH.

In this chapter we present three masking methods enforcing k-anonymity that are well suited for the anonymisation of categorical attributes from a semantic point of view. They are able to build an anonymised dataset that is semantically similar to the original one, thus preserving its analytical utility. To do so, we rely on the notion of *ontology-based semantic similarity*, which enables a semantically-coherent management of categorical data.

An ontology, in Information Science, can be defined as a rigorous and exhaustive organization of some knowledge domain that is usually hierarchical and contains all the relevant entities and their relations. It is used to describe a domain of knowledge and to reason about the properties of the objects in that domain [10]. Ontologies are machine-interpretable so that it can be queried. In this sense, some standard languages have been designed to codify ontologies. They are usually declarative languages based on either first-order logic or on description logic [11]. Thanks to initiatives such as the Semantic Web, which brought the creation of thousands of domain ontologies [12], ontologies have been extensively exploited to compute semantic similarity between entities.

In the remainder of the chapter we describe three semantically grounded masking methods that exploit available ontologies to preserve the semantics of anonymised data. They are based on classic techniques found in Statistical Disclosure Control: Recoding, Microaggregation and Re-sampling. The three methods have been applied to enforce k-anonymity of Electronic Health Records. An empirical comparison of their performance is done in terms of semantic utility, privacy disclosure risk and runtime.

1.1 Problem Formalisation

A structured database consists of n records corresponding to individuals, each one containing m attribute values. In the simplest case, let us take anunivariate dataset with a single categorical attribute with n records. On the contrary to continuous-scale numerical values, categorical attributes take values from a finite set of modalities (e.g., medical diagnoses); hence, they tend to repeat in the different records of the database. To explicitly consider repetitions, we represent the dataset in the form of $V = \{< v_1, \omega_1 >, \ldots, < v_p, \omega_p >\}$, where each $< v_i, \omega_i >$ tuple states the number ω_i of repetitions of each *distinct value* v_i found in V. Note that, typically, p (i.e., the number of distinct values) would be significantly lower than n (i.e., the total amount of records).

Example 1 Given the univariate dataset V with $n = 12$ patient diagnosis: {*asbestosis*,*degenerative disorder*, *amyotrophia*, *myofibrosis*, *asbestosis*, *allergy*, *myofibrosis*, *allergy*, *squint*, *amyotrophia*, *degenerative disorder*, *allergy*}, we represent this dataset as a tuple with $p = 6$ elements:

$$\begin{aligned} V = \{ & < \textit{asbestosis}, \quad 2 >, < \textit{degenerative disorder}, \quad 2 >, \\ & < \textit{amyotrophia}, \quad 2 >, < \textit{myofibrosis}, \quad 2 >, < \textit{allergy}, \quad 3 >, \\ & < \textit{squint}, \quad 1 > \} \end{aligned}$$

This formalisation can be generalised for multivariate datasets with $m > 1$ attributes as follows. Let $MV = \{< \{v_{11}, \ldots, v_{1m}\}, \omega_1 >, \ldots, < \{v_{p1}, \ldots v_{pm}\}, \omega_p >\}$ be the representation of the dataset, where each tuple $\{v_{i1}, \ldots, v_{im}\}$ represents a distinct combination of m attribute values, and ω_i states its number of occurrences (i.e., the frequency).

Example 2 Given the multivariate dataset *MV* with two attributes describing conditions and treatments of a set of $n = 8$ patients: {{*lumbago*, *rehabilitation*}, {*colic*, *antibiotic*}, {*lumbago*, *rehabilitation*}, {*migraine*, *aspirin*}, {*lumbago*, *rehabilitation*}, {*lumbago*, *codeine*}, {*colic*, *hospitalisation*}, {*lumbago*, *codeine*}}, we represent this multivariate dataset as a tuple with $p = 5$ elements:

$$\begin{aligned} MV = \{ & < \{lumbago, rehabilitation\}, 3 >, < \{colic, antibiotic\}, 1 >, \\ & < \{migraine, aspirin\}, 1 >, < \{lumbago, codeine\}, 2 >, \\ & < \{colic, hospitalisation\}, 1 > \} \end{aligned}$$

Given that our goal to fulfil k-anonymity, if ω_i is equal or greater than a given value of k, the corresponding records in this tuple are already k-anonymous since they fulfil desired level of privacy. Hence, the goal of the masking process consists of generating a dataset where $\omega_i \geq k, \forall i$.

By managing the input dataset as formalised above we will be able to improve the computational cost of the algorithms, since they will be a function of p instead of n. Moreover, given that categorical data is characterised by a limited and usually reduced set of modalities, it is expected that $p \ll n$.

2 Semantic Data Recoding

Data recoding, also known as *generalization* [13], is a making method that combines several categories in a new (less specific) one. For continuous attributes, global recoding means replacing an attribute by its discretized version, but the discretization leads very often to an unaffordable loss of information. For categorical attributes the methods rely on hierarchies of terms covering the categorical values observed in the sample, in order to replace a value by another more general one.

Previous works used ad-hoc small VGHs to find suitable generalisations. On the contrary, our ontology-based recoding takes advantage of the wide coverage of ontologies in different domains, such as WordNet or SNOMED CT. We present here a new recoding method capable of mapping categorical attribute values into ontological nodes that do not necessarily represent leaves of a hierarchy. As a result, semantically related concepts can be retrieved by going through the ontological hierarchies to which the value belongs. These ontological hierarchies are designed in a much more general and fine-grained fashion than VGHs and as a result of the consensus between domain knowledge experts rather than of the input data. Since we do not restrict the replacement to strict generalisations but to semantically similar entities, we enable a much wider and semantically-coherent set replacements. To ensure scalability with regards to the ontology size and input data, we bind the space of valid replacements to the set of value combinations that are present in the input dataset. When changing one value of a record for another, we can substitute an element by a taxonomical subsumer (this is the only case covered by the classic generalisation methods) but also with a hierarchical sibling (with the same taxonomical depth) or a specialisation (located at a lower level). In fact, in many situations a specialisation can be more similar than a subsumer,because highly specific concepts belonging to the lowest levels of a hierarchy have less differentiated meanings. As a result, the value change would result in a higher preservation of the semantic of data. This is an interesting characteristic and an improvement over the more restricted data transformations supported by VGH-based generalisation methods.

The proposed method is based on the substitution of all quasi-identifier values of each record with the values of another record [14]. To ensure the scalability of the method and guide the anonymisation process towards the minimisation of information loss, we have designed two heuristics (H) that ensure the selection, at each iteration, of the best set of records to transform:

(H_1) From the input dataset, select the set of records T with the lowest number of repetitions.

(H_2) For each record $t \in T$ find the least distant record v in the input dataset.

The goal of the first heuristic is to start the process with the records that fulfil k-anonymity the least,whereas, the aim of the second heuristic is minimise the information loss resulting from each substitution. To select such substitution, a semantic comparison operator has been defined. As a result of the replacement, quasi-identifier values for both records (the one to anonymise t and the most semantically similar one v) will take the same values and become indistinguishable; therefore, the k-anonymity level for both records will increase.

The recoding algorithm presented in [14] proceeds as follows:

1. Select the tuple t with the minimum number of repetitions.
2. As long as this number is lower than k, the dataset is not k-anonymous. To increase the number of repetitions, select the tuple v that is the least distant to t and that has the lowest amount of repetitions.
3. The original values of t are replaced by the ones in v, increasing, in this manner, their anonymity level because the value in v will have a higher number of repetitions.
4. Go to step 1.

This algorithm stops when in step 2 we have that the tuples with the minimum number of repetitions have a number of repetitions equal or higher than k. Consequently, with this iterative procedure that is applied to each non-anonymous record, the input dataset will fulfil k-anonymity.

In [15] the applicability and quality of the anonymised data has been demonstrated with several tests conducted with real data. Results indicate that, compared with a classical approach based on optimisation of the distribution of the data, ours retains the quality and utility of data better from a semantic point of view. This was studied by means of comparing the results of a clustering process. The partitions generated from the original dataset and the anonymised data are more similar with our semantic method than with classical approaches.

2.1 Semantic Similarity

To compare the categorical values and to calculate the least distant tuples from a semantic perspective, the following similarity function is proposed [16]. This similarity function takes into account both the data distribution as well as the semantic similarity of the terms.

On the one hand, the semantic similarity of the values is estimated by mapping them into ontological concepts. Then, any of the edge-counting measures that can be found in the literature to evaluate the distance between concept pairs according to the length of the path connecting the two concepts in the ontology can be used [17]. The main advantage of the edge-counting measures is their simplicity and low computational cost. However, they require rich and consistent ontologies to work properly.

On the other hand, to consider also the distribution of data during the comparison of attribute values, their frequency of appearance is also taken into account. Since each distinct value v_i appears ω_i times in the dataset, we propose to count the semantic distance between a given value v_i and a base value b as many times as indicated by its frequency of appearance ω_i. Since this is equivalent to multiplying the semantic distance by the frequency, ω_i acts as a weighting factor. In that way, the accumulated distances resulting from grouping together a base value b and all the records with the value v_i can be minimised. Moreover, since all repetitions ω_i of v_i are treated as a unit, sets of identical records will be grouped together, obtaining more cohesive groups.

Formally, the comparison operator used to group records with respect to a base value b is defined as follows.

Definition 1 The weighted semantic distance (*wsd*) between anunivariate reference value b and a univariate set of records $< v_i, \omega_i >$ is defined as:

$$wsd\,(b, < v_i, \omega_i >) = \omega \cdot sd(b,\ v_i) \tag{1}$$

This measure can be generalised to multivariate data as follows:

Definition 2 The distance between a multivariate reference value with m attributes $\{b_1, \ldots, b_m\}$ and a multivariate set of records $< \{v_{i1}, \ldots, v_{im}\}, \omega_i >$ is defined as the *average* of the weighted semantic distances of the individual attribute values:

$$wsd\,(\{b_1, \ldots, b_m\}, < \{v_{i1}, \ldots v_{im}\}, \omega_i >) = \sum_{j=1}^{m} \frac{wsd\left(b_j, < v_{ij}, \omega_i >\right)}{m} \tag{2}$$

Note that this makes that the computational cost of our algorithms uniquely depend on the number of different tuples (p), unlike related works, which depend on the total size of the dataset (n) and on the depth and branching factor of the hierarchy (which represent an exponentially large generalisation space of substitutions to evaluate).

3 Semantic Microaggregation

Among the plethora of anonymisation methods, *microaggregation* stands as a natural approach to satisfy k-anonymity in statistical databases [18]. It builds clusters of at least k original records according to a similarity function; then, each record of each cluster is replaced by the centroid of the cluster to which it belongs. As a result, each combination of values is repeated at least k times and, hence, the masked dataset becomes k-anonymous. The goal of microaggregation is to find the partition that minimises the information loss. Because the search for the optimal partition when considering multivariate data is NP-hard [19], sub-optimal heuristic methods have been proposed. One the most popular ones is the MDAV (Maximum Distance

Average Vector) method [20], because it provides high quality aggregations without being constrained by some configuration parameters, as other methods do [21]. Our proposal focuses on adapting this method to categorical data.

The MDAV method is based on generating clusters of at least k elements. First, the centroid of the dataset is calculated and the most distant object r is selected by means of a distance measure appropriate for the type of data. Then, a cluster is constructed with the $k-1$ nearest objects to r. After that, the most distant record s to r is selected and a new cluster is constructed. The whole process is repeated until no objects remain ungrouped. As a result of the microaggregation process, all clusters have a fixed size of k, except the last cluster that may have a cardinality between k and $2k-1$, because the initial number of records may not be divisible by k. Finally, all the elements in each cluster are replaced by the centroid of the cluster, becoming k-anonymous.

The MDAV method, however, presents some limitations that may hamper the utility of categorical data.

- The fact of relying on fixed-size clusters is a hard restriction that hampers the quality of the clusters in terms of cohesion. A low cohesion increases the information loss resulting from the replacement of the individual records by the cluster centroid [22]. The possibility of varying the size of the clusters ensuring a minimum cardinality of k to fulfil k-anonymity would be preferable because it allows a better adaptation of the clusters to the data distribution. This is especially relevant for *categorical* data like job or city of living because, due to their discrete nature, modalities tend to repeat and, hence, it would be desirable to put as many repetitions as possible into the same cluster to maximise its cohesion.
- The results are influenced by the two operators needed during the microaggregation: the *distance measure*, used to compare records and centroids, and the *centroid construction,* needed to calculate the global centroid at each iteration and to select the representative record for each cluster. Since arithmetic functions cannot be applied to this kind of data, a straightforwardway to apply MDAV to categorical data consists on using Boolean equality/inequality operators [18, 23] or to use the common abstraction of a set of values in an ontology as the centroid [24].

In our proposal, some algorithmic and design modifications are made by considering the distributional characteristics of categorical attributes with the purpose of minimising the information loss. Two main aspects have been considered: (1) the interpretation of the semantics of non-numerical values during the whole microaggregation process, and (2) the consideration of the distribution categorical attributes to define adaptive-sized clusters, producing more cohesive results while fulfilling k-anonymity. The goal is to aggregate data into highly cohesive clusters, in order to minimise the information loss resulting from the masking process. The proposed modifications are the following [25]:

- Adaptive microaggregation: we propose a modification of the MDAV algorithm that, while ensuring the k-anonymity property, creates clusters of different sizes according to the data distribution. The algorithm will put all the records that have

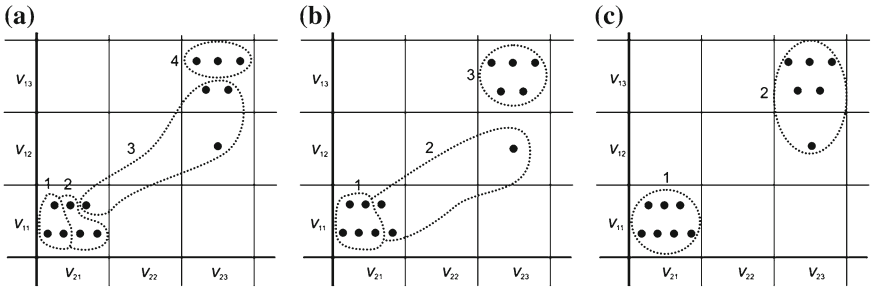


Fig. 1 An example of microaggregation with $k = 3$. **a** Fixed-sized clustering. **b** Variable-sized clustering with a maximum size of $2k - 1$. **c** Adaptive clustering without maximum size restriction

the same values in the same cluster, while ensuring that the cluster has, at least, k elements to fulfil the k-anonymity property. In this manner, the clustering construction process is guided by the data distribution, as illustrated in Fig. 1.

- Semantic distance: given that categorical data should be interpreted according to their underlying semantics, we propose to use the same weighted semantic similarity operator define above (Eq. 2) to guide the cluster construction process. It considers both the meaning of the value according to the background ontology and the distribution of those values.
- Semantic centroid: the construction of an appropriate and representative centroid is crucial to guide the microaggregation process and to minimise the information loss. We proposed in [25] a new procedure to calculate the centroid of multivariate categorical datasets by exploiting ontologies as well as considering the data distribution.

As a result of incorporating this adaptive behaviour during the clustering construction, the SA-MDAV (*Semantic Adaptive MDAV*) is presented [26]:

1. Compute the centroid *X*of all tuples in input data.
2. Select the most distant tuple r to the centroid X and the most distant tuple s to r.
3. Form a cluster with the tuple r and calculate the centroid C of the cluster.
4. Add to this cluster the closest tuples to centroid C, recalculating the centroid whenever a tuple is added, until at the number of records is at least k.
5. Form a cluster with the tuple s and do the same process as for r.
6. Remove the added tuples from input data and repeat the entire process until no tuples remaining in the dataset.
7. Replace each tuple by the centroid of the cluster that it belongs to.

We evaluate this method over two different datasets with categorical (details in [15, 16]). Results proved that SA-MDAV, even though being heuristic and subject to sub-optimal choices to preserve its scalability, improves related works by a considerable margin, both when considering the absolute information loss and also when evaluating the balance between information loss and disclosure risk.

3.1 The Centroid of Categorical Values

Centroid calculus for numerical data relies on standard averaging operators (e.g. arithmetic mean) [2]. However, the accurate centroid calculus for non-numerical data is challenging due to the lack of semantic aggregation operators and the necessity of considering a discrete set of centroid values. Related works propose methods to compute centroids for non-numerical data *either* relying on the distributional features of data, where the centroid is the *modal* value [23], *or* on background semantic, where the centroid is the term that generalises all aggregated values in a taxonomy [24]. Since only one dimension of data (distribution or semantics) is considered, both approaches result in suboptimal results [25].

In this section, we introduce a centroid calculation method for multivariate non-numerical data that considers, in an integrated manner, *both* semantics and distribution of data. To obtain accurate centroids, ontologies are exploited not only to semantically compare terms, as proposed in the previous section, but also to retrieve the centroid candidates.

The first issue concerns the search space of centroid candidates (c). Since c must be necessarily a discrete value, some approaches (like the ones based on taking the modal value of a sample [23]) bound the set of possible candidates to those values appearing in the input dataset. Hence, the centroid accuracy would depend on the granularity and suitability of the input data. So, we extend the centroid search space to all terms of the taxonomy related to the input data. Centroid candidates will be all input terms together with their taxonomical ancestors. For semantic comparison, the weighted semantic distance defined in the previous section is used.

Formally, let us take $V = \{< v_1, \omega_1 >, \ldots, < v_p, \omega_p >\}$ as an input dataset with a single non-numerical attribute. The first step maps the terms v_i of the set V in a background ontology and extracts the minimum hierarchy H that taxonomically models all v_i values. All terms in H, which include both values in V and their taxonomical ancestors, are considered as centroid candidates. The centroid will be the term c in H that minimises the weighted semantic distance (Eq. 2) to all v_i in V. Note that, in this case, each centroid candidate c acts as the base value in Eq. (2).

Definition 3 The centroid of a set of non-numerical values v_i in V is defined as the term c_j that minimises the weighted semantic distance *wsd* with respect to all the values v_i in the space V.

$$centroid\,(V) = \{\arg\min_{\forall c_j \in H}(\sum_{i=1}^{p} wsd(c_j, < v_i, \omega_i >))\} \tag{3}$$

4 Semantic Resampling

The third masking method is based on dataresampling. Resampling was originally proposed for protecting tabular data, but later it has been used for microdata [27]. Although resampling has not gained as much research attention as other masking

algorithms like microaggregation [22], it has demonstrated to be fast, which is an interesting feature when dealing with large-scale datasets and to retain more utility than other methods for numerical data [28, 29].

Briefly, being n the number of records in the dataset, the resampling method takes t samples with replacement (i.e. values can be taken more than once). Each sample is sorted in increasing order. Then, the masked dataset is obtained by taking, as first value, the average of the first values of all samples, as second value, the average of the second values of all samples, and so on.

Comparative studies [28, 29] show that Heer's resampling achieves a high utility preservation with respect to other masking techniques, but with a slightly higher disclosure risk. This is related to the fact that, unlike other masking methods [22–24], the Heer's approach was designed without considering k-anonymity (formalized years later in [13]). Hence, resampled results cannot guarantee an a priori level of privacy.

A new version of resampling is presented here, named SkResampling [30]. The new resampling method fulfils k-anonymity while it is also able to deal with non-numerical data from a semantic perspective. Two issues not considered in previous works.

The SkResampling algorithm is focused on minimizing the information loss when masking categorical data while ensuring the fulfilment of k-anonymity. It is based on the Heer's resampling method [27] with the following modifications:

- *k-anonymous resampling*: the original sampling method has been modified so that masked records fulfil k-anonymity.
- *Semantic resampling of categorical data*: in order to semantically interpret non-numerical data during the resampling process, we have applied semantic operators, such as the weighted semantic similarity function and a sorting operator presented below.

Let D be the input dataset, k is the desired level of k-anonymity and n is the number of records in D. The algorithm proceeds as follows:

1. Create k set of n/k records, by a sampling procedure without replacement (i.e. each record is taken only once). The aim it that k records can be replaced by their average in a later stage and become k-anonymous.
2. Sort each of these samples with the same semantic criterion.
3. Create P_i ordered sets with the records at the ith position of all sorted samples. The idea is that, by sorting the samples, similar records appear at similar positions of different samples.
4. The anonymised dataset is obtained by replacing all records of each P_i by the centroid of P_i.

With this algorithm P_i contains at least k records that are all substituted by the same centroid, hence, we guarantee that the masked dataset is k-anonymous. Note that, in step 1, when taking n/k records per sample, the remaining $n \bmod k$ records should be also treated. These records are added to the set with the closest centroid (in step 3), recalculating the centroid before executing step 4.

In comparison with other methods, resampling is faster since the sampling process is done randomly, which makes it especially suitable for very large datasets. In contrast, this randomness may negatively influence the information loss. The evaluation results sustain our hypothesis, since the method is able to minimise the information loss in comparison with non-semantic approaches [16].

4.1 Sorting Non-numerical Data

In the resampling method, a sorting operator is required. In [16] we proposed a procedure to arrange the records with non-numerical data according to their semantic similarity.

Sorting categorical values is not straightforward since, in general, they are not ordinal (i.e. they do not take values in an *ordered* range). In this section, we detail a sorting algorithm for categorical data.

To sort a set of values, a reference point is needed so that values could be arranged according to their similarity/distance to that reference. Numerically, this is done according to the max/min value (i.e. the most extreme value) of the set. To define a sorting procedure for categorical data, we also rely on the notion of the most extreme value, which corresponds to the one that, globally, is the *most distant* to all other values (conceptually, this is the opposite of the centroid as computed in Sect. 3.1). Once this reference value is obtained, other values are sorted by iteratively picking those that are least distant to that extreme value.

To set the reference value/record as well as to compare it to other elements in the set, while considering both the semantic and distributional features of data, we rely on the weighted semantic distance and the centroid calculus procedures explained before.

5 A Comparative Study in the Medical Domain

This section compares the three methods detailed above when applied to the anonymisation of real medical data. In particular, we address the problem of protecting the privacy of Electronic Health Records of patients. This kind of dataset contains confidential information regarding clinical outcomes that should not be disclosed. At the same time, the analysis of the health care experience captured in clinical databases is very important because it may lead to improved continuity in patient assessment, improved treatment, avoidance of adverse drugs reactions, and in ensuring that people at risk receive appropriate support services [31, 32]. Thus, utility of anonymised data is critical.

Moreover, due to the importance of terminology in clinical assessment, the medical domain has been very prone to the development of large and detailed ontologies such SNOMED CT [33, 34], which is used in our tests to protect medical terms.

Table 1 Example of clinical dataset provided by OSHPD

ID	Age range	Patient ZIP code	Principal diagnosis cause of admission	Other condition that coexist at the time of admission
*	50–54	916**	Abstinent alcoholic	Metabolic acidosis due to salicylate
*	65–69	913**	Infected spinal fixation device	Uric acid renal calculus
*	65–69	903**	Aneurysm of thoracic aorta	Cardiac oedema
*	>=85	902**	Fibroma of ovary	Chronic osteoarthritis
*	30–34	917**	Appendicitis	Severely obese

5.1 The Dataset

The evaluation has been carried out over a database containing inpatient information provided by the California Office of Statewide Health Planning and Development (OSHPD) and collected from licensed hospitals in California. Specifically, we used the patient discharge dataset corresponding to the 4th quarter of 2009 for the hospital with the largest amount of records (i.e., Cedars Sinai Medical Center, Los Angeles County).

Prior to publication, the OSHPD has masked or removed some attributes that resulted in unique combinations of certain demographic variables (see an example on the first three columns of Table 1). For evaluation purposes, two categorical attributes were considered in our tests: *principal diagnosis* and *other conditions* of the patient at the time of the admission (i.e., $m = 2$), which are stored as ICD-9 codes in the original data file. After removing records with missing information, a total of 3,006 individual records is available for testing (i.e., $n = 3,006$). A total of 2,331 different combinations of values (i.e., $p = 2,331$ tuples) can be found, from which a significant amount (2,073) are unique.

5.2 Comparing Anonymisation Methods

The three algorithms have been compared under the perspectives of information loss, disclosure risk and runtime. The information loss (L) measure estimates the utility of anonymised dataset by quantifying the *semantic information loss* caused by replacing original values by their masked versions. It is measured as the *Sum of Squared Errors* (SSE). To measure the information loss from a *semantic* perspective, we compare the categorical values using the semantic weighted distance defined in Eq. (2) on the basis of the knowledge available in the SNOMED CT ontology.

$$L = \sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{m} sd(r_{ij}, r_{ij}^{A})}{m} \right)^{2} \tag{4}$$

where n is the number of records in the dataset, each one composed by m attributes, r_{ij} is the original value of the jth attribute of the ith record and r_{ij}^{A} denotes its masked version.

On the other hand, the disclosure risk is calculated by means as the percentage of correct linkages between the original and masked datasets. The Record Linkage (RL) is assessed also by the same semantic similarity measure and SNOMED CT as knowledge base. The specific record linkage method is detailed in [35].

To evaluate the balance score between the information loss and the disclosure risk of the different methods, we use the score function (Eq. 5). The lower the score is, the higher the quality of the method because both low information loss and low disclosure risk are achieved.

$$score = \alpha L + (1 - \alpha) RL \tag{5}$$

First, as shown in Fig. 2, we consider an equal balance between the data utility and the disclosure risk, that is, $\alpha = 0.5$. The conclusion is that the recoding method provides the best results for low k levels ($k <= 5$) and the microaggregation method works better for high k levels ($k > 5$). Thus, depending on the level of k-anonymity it would be convenient to choose a method or another. It is also relevant to note that the score is maintained almost constant as k-values grow, stating that the quality of the methods scales well as the privacy requirements increase. On the other hand, resampling provides the worst balance between information loss and disclosure risk.

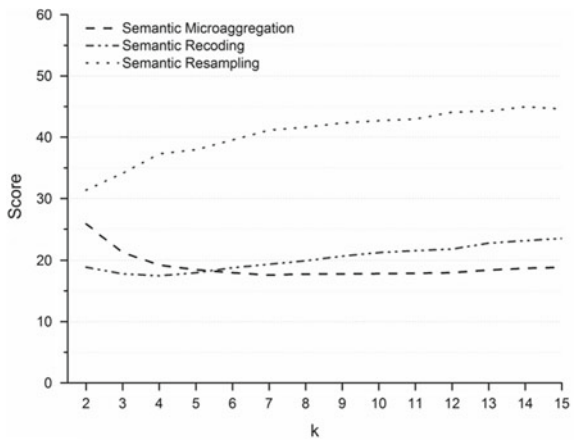


Fig. 2 Score with an equilibrated balance ($\alpha = 0.5$) between information loss and disclosure risk

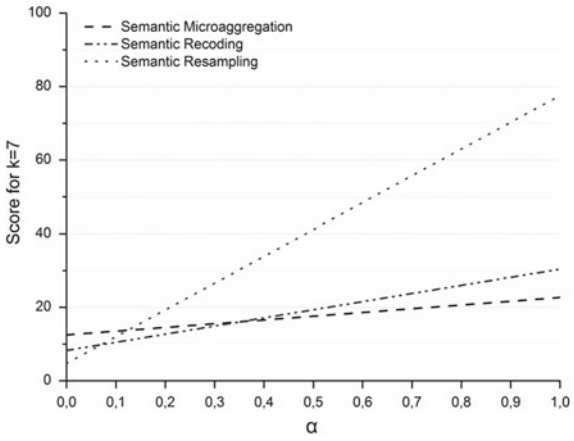


Fig. 3 Score values when varying the trade-off between information loss and disclosure risk

We have also studied the behaviour of the overall score when varying the parameter α between 0 to 1. With $\alpha = 0$, the score is based solely on the disclosure risk measure, while with $\alpha = 1$, the score is based only on the information loss. In this analysis, an intermediate level of anonymity ($k = 7$) has been fixed.

As it can be seen in Fig. 3, the recoding method achieves the best results for low values of α whereas themicroaggregation method obtains the best results for high values of α. This means that if we need to give more importance to information loss with an adequate level of disclosure risk, the microaggregation method would be more suitable. On the contrary, if we give more weight to disclosure risk with a moderate level of information loss, the recoding method would be preferable. The resampling method obtains the best result only when data utility is not taken into account.

Finally, the runtime of the anonymisation algorithms is a relevant feature to consider when resources are limited, such as in EHRs, which are likely to contain large amounts of data. Figure 4 shows the comparison for the three SDC methods executed on a 2.4 GHz Intel Core processor with 4 GB RAM. The fastest method is *resampling*, with an almost negligible runtime. *Microaggregation* is the slowest for values of k below 4, whereas it surpasses *recoding* for higher k-values, with an almost inverse logarithmic shape.

Given the above results, the semantically-grounded *microaggregation* method seems the best approach to anonymise data when the meaning of original data should be preserved as much as possible with a moderate level of disclosure risk. Moreover, it is especially efficient for high k-anonymity values. *Recoding* would be only considered if very low k-anonymisation levels are required (being more computationally efficient than *microaggregation*) or when analyses to be performed over masked data will be focused solely on data distribution rather than on their semantics. Finally, only when input EHRs are so large to be non-computationally feasibly anonymised

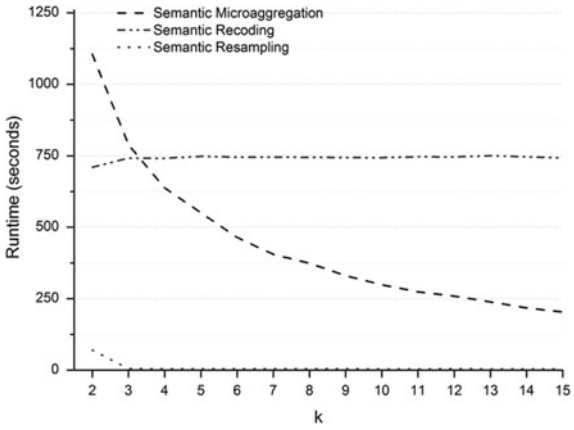


Fig. 4 Runtime (in seconds) for different levels of *k*-anonymity

by means of *microaggregation* or *recoding* methods, the *resampling* method could be considered thanks to its high efficiency, even at the expenses of a higher information loss.

6 Conclusions and Future Work

In this chapter, we presented some methods for anonymisation for categorical attributes, showing that data semantics plays acrucial rolein order to retain the analytical utility. Given the importance of data semantics in anonymisation tasks, we have studied the definition of new operators needed in masking methods. These operators exploit the semantics provided by ontologies to enable a coherent comparison, aggregation and sorting of categorical data and without neglecting the distribution of the values.

The three masking methods apply different perturbation methods in order to achieve *k*-anonymity, namely by recording original values, by microaggregation of semantically similar records (that are substituted by a common centroid) or by sampling and sorting the original data into small sets (also replaced by their centroid).

The three methods have been evaluated with a real dataset from different perspectives, concluding that each method has its own advantages and limitations. We can also conclude that well-defined general purpose semantic structures, as ontologies, are a good source of information to interpret the semantics of terms and their use is crucial to retain the utility of data during the anonymisation process.

This work shows that it is possible to use ontologies with traditional masking methods if some operators are adapted. The three operators presented here appear in other anonymisation techniques available in the literature, which could be easily adapted to deal appropriately with categorical data.

As future work, it would be interesting to study how the different methods behave with other ontologies with different sizes and granularities. The possibility of combining several ontologies as background knowledge could be also considered. Regarding the privacy model, research on the application of the proposed semantic framework to other models and methods can be done. Related with the privacy model, recently, a more robust privacy model like *differential privacy* can be considered. *Differential privacy* [36] ensures that released data are insensitive to any individual's data. Hence, individual data remains uncertain for an attacker. Finally, we would also consider the adaptation of other numerical operators such as variance or co-variance for categorical values by means of ontologies.

Acknowledgments This work has been supported by the Spanish Ministry of Science and Innovation (through projects ICWT TIN2012-32757, ARES-CONSOLIDER INGENIO 2010 CSD2007-00004 and BallotNext IPT-2012-0603-430000) and by the Government of Catalonia under grants 2009 SGR 1135 and 2009 SGR-01523. Dr. Martínez was supported with research grants by the Universitat Rovira i Virgili and Ministerio de Educación y Ciencia (Spain).

References

1. Willenborg, L., de Waal, T.: Elements of Statistical Diclosure Control. Lecture Notes in Statistics, vol. 155. p. 261. Springer, New York (261)
2. Domingo-Ferrer, J.: A Survey of inference control methods for privacy-preserving data mining. In: Aggarwal, C.C., Yu, P.S. (eds.) Privacy-Preserving Data Mining, pp. 53–80. Springer, US (2008)
3. Jin, X., Zhang, N., Das, G.: ASAP: eliminating algorithm-based disclosure in privacy-preserving data publishing. Inf. Syst. **36**(5), 859–880 (2011)
4. Herranz, J., et al.: Classifying data from protected statistical datasets. Comput. Secur. **29**(8), 875–890 (2010)
5. Oliveira, S.R.M., Zaïane, O.R.: A privacy-preserving clustering approach toward secure and effective data analysis for business collaboration. Comput. Secur. **26**(1), 81–93 (2007)
6. Shin, H., Vaidya, J., Atluri, V.: Anonymization models for directional location based service environments. Comput. Secur. **29**(1), 59–73 (2010)
7. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl. Based Syst. **10**(5), 557–570 (2002)
8. Aggarwal, C.C., Yu, P.S.: Privacy-Preserving Data Mining: Models and Algorithms. Springer Publishing Company, Incorporated, Berlin (2008)
9. Torra, V.: Towards knowledge intensive data privacy. In: Proceedings of the 5th International Workshop on Data Privacy Management, and 3rd International Conference on Autonomous Spontaneous Security, Springer, Athens, Greece (2011)
10. Guarino, N.: Formal, ontology and information systems. In: 1st International Conference on Formal Ontology in Information Systems. IOS Press, Trento, Italy (1998)
11. Gomez-Perez, A., Fernandez-Lopez, M., Corcho, O.: Ontological Engineering, 2nd Printing. Springer, New York (2004)
12. Ding, L. et al.: Swoogle: a search and metadata engine for the semantic web. In: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management. ACM, Washington, D.C., USA (2004)
13. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, SRI Computer Science Laboratory (1998)

14. Martínez, S., et al.: Privacy protection of textual attributes through a semantic-based masking method. Inf. Fusion **13**(4), 304–314 (2011)
15. Martínez, S.: Ontology based semantic anonimisation of microdata. Universitat Rovira i Virgili. PhD. Thesis (2013). http://www.tdx.cat/bitstream/handle/10803/108961/Tesi.pdf?sequence=1
16. Martínez, S., Sánchez, D., Valls, A.: A semantic framework to protect the privacy of electronic health records with non-numerical attributes. J. Biomed. Inform. **46**(2), 294–303 (2013)
17. Rada, R., et al.: Development and application of a metric on semantic nets. IEEE Trans. Syst. Man Cybern. **19**(1), 17–30 (1989)
18. Domingo-Ferrer, J., Torra, V.: Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Min. Knowl. Discov. **11**(2), 195–212 (2005)
19. Oganian, A., Domingo-Ferrer, J.: On the complexity of optimal microaggregation for statistical disclosure control. Stat. J. United Nations Econ. Comm. Eur. **18**(4), 345–353 (2001)
20. Hundepool, A. et al.: μ-ARGUS version 3.2 software and user's manual. Statistics Netherlands, Voorburg NL (2003). http://neon.vb.cbs.nl/casc://neon.vb.cbs.nl/casc
21. Domingo-Ferrer, J., et al.: Efficient multivariate data-oriented microaggregation. VLDB J. **15**(4), 355–369 (2006)
22. Domingo-Ferrer, J., Mateo-Sanz, J.M.: Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans. Knowl. Data Eng. **14**(1), 189–201 (2002)
23. Torra, V.: Microaggregation for categorical variables: a median based approach. In: Domingo-Ferrer, J., Torra, V. (eds.) Privacy in Statistical Databases, pp. 518–518. Springer, Berlin (2004)
24. Abril, D., Navarro-Arribas, G., Torra, V: Towards semantic microaggregation of categorical data for confidential documents. In: Proceedings of the 7th International Conference on Modeling Decisions for Artificial Intelligence. Springer, Perpignan, France (2010)
25. Martínez, S., Valls, A., Sánchez, D.: Semantically-grounded construction of centroids for datasets with textual attributes. Knowl. Based Syst. **35**, 160–172 (2012)
26. Martínez, S., Sánchez, D., Valls, A.: Semantic adaptive microaggregation of categorical microdata. Comput. Secur. **31**(5), 653–672 (2012)
27. Heer, G.R.: A bootstrap procedure to preserve statistical confidentiality in contingency tables. In: International Seminar on Statistical Confidentiality. Eurostat, Luxembourg (1993)
28. Herranz, J., Nin, J., Torra, V.: Distributed privacy-preserving methods for statistical disclosure control data privacy management and autonomous spontaneous security. Int. Sci. **5939**, 33–47 (2010)
29. Karr, A.F., et al.: A framework for evaluating the utility of data altered to protect confidentiality. Am. Stat. **60**, 224–232 (2006)
30. Martínez, S., Sánchez, D., Valls, A.: Towards k-anonymous non-numerical data via semantic resampling. In: Greco, S. et al. (eds.) Information Processing and Management of Uncertainty in Knowledge-Based Systems, Catania, Italy (2012)
31. Elliot, M., Purdam, K., Smith, D.: Statistical disclosure control architectures for patient records in biomedical information systems. J. Biomed. Inform. **41**(1), 58–64 (2008)
32. Malin, B., Sweeney, L.: How (not) to protect genomic data privacy in a distributed network: using trail re-identification to evaluate and design anonymity protection systems. J. Biomed. Inform. **37**(3), 179–192 (2004)
33. Spackman, K.A., Campbell, K.E., Cote, R.A.: SNOMED RT: a reference terminology for health care. In: Proceedings of AMIA Annual Fall Symposium, pp. 640–644 (1997)
34. Nelson, S.J., Johnston, D., Humphreys, B.L.: Relationships in medical subject headings. In: Relationships in the Organization of Knowledge, pp. 171–184. K.A. Publishers, New York (2001)
35. Martínez, S., Sánchez, D., Valls, A.: Evaluation of the disclosure risk of masking methods dealing with textual attributes. Int. J. Innovative Comput. Inf. Control **8**(7(A)), 4869–4882 (2012)
36. Dwork, C.: Differential privacy. In: ICALP, Springer (2006)

Contributions on Semantic Similarity and Its Applications to Data Privacy

Montserrat Batet and David Sánchez

Abstract Semantic similarity aims at quantifying the resemblance between the meaning of textual terms. Thus, it represents the corner stone of textual understanding. Given the increasing availability and importance of textual sources within the current context of Information Societies, a lot of attention has been put in recent years in the development of mechanisms to automatically measure semantic similarity and to apply them to tasks dealing with textual inputs (e.g. document classification, information retrieval, question answering, privacy-protection, etc.). This chapter offers describes and discusses recent findings and proposals published by the authors on semantic similarity. Moreover, it also details recent works applying semantic similarity to privacy protection of textual data.

1 Introduction

The enormous development of the World Wide Web and the Information Societies has made available large amounts of electronic resources. Because many of these resources are of a textual nature, a great interest has been shown in recent years about the automated understanding of textual contents. The counter stone of textual understanding is the assessment of the *semantic similarity* between textual entities (e.g. words, phrases, sentences or documents).

Semantic similarity aims at assessing a score that quantifies the resemblance between the meanings of the compared entities, so that algorithms relying on such

M. Batet (✉) · D. Sánchez
Department of Computer Engineering and Mathematics,
UNESCO Chair in Data Privacy, Universitat Rovira i Virgili,
Av. Països Catalans, 26, 43007 Tarragona, Catalonia, Spain
e-mail: montserrat.batet@urv.cat

D. Sánchez
e-mail: david.sanchez@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_8

assessment (e.g. classification, clustering, etc.) can seamlessly manage textual information from a numerical perspective. Because data semantics is an inherently human feature, semantic similarity measures proposed in the literature exploit one or several human-tailored information or knowledge sources, which are exploited under different theoretical principles. According to such features and principles, the first part of this chapter offers a classification and a comparative discussion of recent findings and proposals published by the authors on semantic similarity.

Due to its core importance and the need of dealing with textual inputs, semantic similarity has been applied in recent years in a variety of tasks, which include natural language processing, information management and retrieval, textual data analysis and classification or privacy-protection. Regarding the latter, in which privacy protection methods obfuscate sensitive information in order to guarantee the fundamental right to privacy of individuals while retaining a degree of data utility [1], data semantics are of utmost importance when dealing with textual inputs: they influence both the risk of disclosing confidential information due to semantic inferences and the utility of the protected output understood as the preservation of data semantics [2, 3]. Privacy protection mechanisms have been usually proposed for numerical inputs, thus focusing on the distributional and statistical features of data. However, neglecting data semantics hampered their applicability and accuracy with textual inputs. Fortunately, in recent years, there has been a growing interest in applying semantic technologies and, particularly, the notion of semantic similarity to offer a semantically-coherent and utility-preserving protection of textual data. The remainder of this chapter details some recent works focusing on such direction.

2 Semantic Similarity

A plethora of semantic similarity approaches are currently available in the literature. This section offers a classification ofthe proposed approaches according to the theoretical principles on which they rely, highlighting their advantages and shortcomings under the dimensions of accuracy, applicability and dependency on external resources (which are summarised in Table 1). Recent findings reported by the authors on each of the approaches are described in more detail.

Table 1 Comparison between similarity calculus paradigms

Measure type	Advantages	Drawbacks
Edge-counting	Simple	Low accuracy
Feature-based	Exploit all taxonomic ancestors to improve the accuracy	A detailed ontology is required
Extrinsic IC-based	Accurate	Suitable tagged corpora is needed
Intrinsic IC-based	Do not require corpora	A detailed ontology is required
1st order co-occurrence	No ontology is required	Compute relatedness rather than similarity
2nd order co-occurrence	Evaluate related terms that do not directly co-occur	Complexity

2.1 Ontology-Based Measures

Ontologies, which have been extensively exploited to compute semantic similarity, define the basic terminology and semanticrelationships comprising the vocabulary of a topic area [4]. From a structural point of view, an ontology is composed by disjoint sets of *concepts, relations, attributes* and *data types* [5, 6]. In an ontology, concepts are organised in one or several *taxonomies* and are linked by means of transitive *is-a* relationships (taxonomical relationships). Multiple inheritance (i.e. the fact that a concept may have several hierarchical ancestors or subsumers) are usually included.

Ontology-based measures estimate the similarity of two concepts according to the structured knowledge offered by ontologies. These measures can be classified into *Edge-counting measures*, *Feature-based measures* and *measures based on Information Content*.

2.1.1 Edge-Counting Measures

They evaluate the number of semantic links (typically is-a relationships) separating the two concepts in the ontology [7–10]. In general, edge-counting measures are able to provide reasonably accurate results when a detailed and taxonomically homogenous ontology is used [8]. They have a low computational cost (compared to approaches relying on textual corpora) and they are easily implementable and applicable.

However, they just evaluate the shortest taxonomical path between concept pairs as evidences of distance (i.e. the opposite to similarity). For example, in Fig. 1 the shortest path length between the concepts *Surfing* and *Sunbathing* is 2 (*Surfing*—

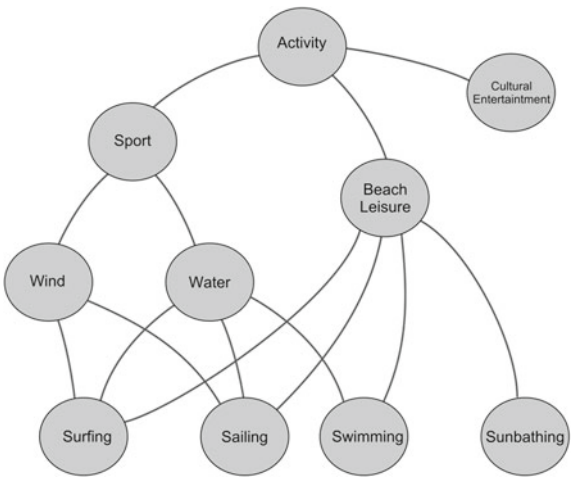


Fig. 1 Sample taxonomy about sports

Beach Leisure—Sunbathing). This is a drawback, because many ontologies (e.g. WordNet, SNOMED-CT or MeSH) incorporate multiple taxonomical inheritance that would result in several taxonomical paths connecting concept pairs, as shown in Fig. 1. These paths represent explicit knowledge that is omitted by edge-counting methods [11]. Because of their simplistic design, edge-counting measures usually provide a lower accuracy than other ontology-based methods [11].

2.1.2 Feature-Based Measures

They rely on the amount of overlapping ontological features (e.g. taxonomic ancestors, concept descriptions, etc.) between the compared concepts [12–14]. Feature-based measures exploit more semantic evidences than edge-counting approaches, evaluating both the commonalities and the differences ofconcepts (e.g. common and different taxonomical ancestors). Since the additional knowledge helps to better differentiate concept pairs, they tend to be more accurate than edge-based measures [12].

However, since some feature-based approaches rely on semantic features other than taxonomical, like non-taxonomic relationships (e.g. meronymy) or concept descriptions (e.g. synsets, glosses, etc.), these measures can only be applied to a subset of the available ontologies, in which this kind ofknowledge is available. In fact, domain ontologies often model semantic featuresrather than taxonomical relationships [15]. Another issue that hampers the applicability of feature-based measures as general purpose solutions is the fact that many of them [13, 14] incorporate ad-hoc weighting parameters that balance the contribution of each semantic feature.

To tackle these problems, in [11, 12], a coherent integration of taxonomic features is proposed, thus avoiding the need of weighting parameters while retaining a high accuracy. The approach in [12] assesses semantic similarity as a function of the amount of common and non-common taxonomic subsumers of concepts. Concretely, the similarity between two concepts c_1 and c_2 is measured according to the inverse non-linear ratio between their disjoint subsumers, as an indication of distance; this value is normalised by the total taxonomic subsumers of the concepts, because concept pairs that have many generalisations in common should be less distant than those sharing a smaller amount [12]. Formally, the semantic similarity measure is defined as (1), where $T(c_i)$ is the set of subsumers of the concept c_i, including itself.

$$sim\,(c_1, c_2) = -\log\left(1 + \frac{|\,T(c_1) \cup T(c_2)| - |T(c_1) \cap T(c_2)\,|}{|T(c_1) \cup T(c_2)\,|}\right) \tag{1}$$

According to the sample taxonomy in Fig. 1, the number of disjoint subsumers between *Surfing* and *Sunbathing* counting themselves is 5 (*Surfing*, *Water*, *Wind*, *Sport*, *Sunbathing*), whereas the their total number of subsumers is 7 (*Surfing*, *Water*, *Wind*, *Sport*, *Sunbathing*, *Beach Leisure*, *Activity*). This results in a similarity value of $-\log_2(1 + (5/7)) = -0.78$.

2.1.3 Measures Based on Information Content

These measures rely on quantifying the amount of information (i.e. Information Content (IC)) that concepts have in common [16–18]. The IC of a concept states the amount of information provided by the concept when appearing in a context. On the one hand, the commonality between the concepts to compare is assessed from the taxonomic ancestors they have in common (which is referred as the Least Common Subsumer, LCS). For example, in Fig. 1, the LCS between Swimming and Sunbathing is Beach Leisure. On the other hand, the informativeness of concepts is computed either extrinsically from the concept occurrences in a corpus [16–18] or intrinsically, according to the number of taxonomical descendants and/or ancestors modelled in the ontology [19, 20].

In classical approaches [16–18] IC is computedextrinsically as the inverse of the appearance probability of a concept c in a corpus (2). Thus, infrequent terms are considered more informative than common or general ones.

$$IC(c) = -\log p(c) \tag{2}$$

However, textual corpora contain terms whereas ontologies model concepts, and, hence, a proper disambiguation and conceptual annotation of each word found in the corpus is required in order to accurately compute concept appearance probabilities [16]. Moreover, corpora should be large and heterogeneous enough to provide robust probabilities and avoid data sparseness (i.e. the fact that there are not enough data to extract valid conclusions about the distribution of terms). In addition, IC-based measures require that the probability of appearance of terms monotonically increases as one moves up in the taxonomy; thus, a concept is coherently considered more informative than any of its taxonomical ancestors and less informative than its descendants [16]. This requirement is fulfilled by computing the probability of appearance of a concept as the probability of the concepts and of any of its specialisations in the given corpus (3).

$$p(c) = \frac{\sum_{w \in W(c)} appearances(w)}{N} \tag{3}$$

where $W(c)$ is the set of words in the corpus whose senses are subsumed by c, and N is the total number of corpus words.

As a result, the background taxonomy must be as complete as possible (i.e. it should include most of the specialisations of a specific concept) to provide reliable results [21]. If either the ontology or the corpus changes, probability re-computations need to be recursively executed for the affected concepts. Scalability problems due to the need of manual annotation of corpora required to minimise language ambiguity hamper the applicability and accuracy of these measures [22].

To overcome these limitations, in recent years, some authors have proposed computing IC in an intrinsic manner by using only the knowledge structure modelled in an ontology [19, 23, 24]. These works assume that the taxonomic structure of ontologies

is organised in a meaningful way, according to the principles of cognitive saliency [25]: concepts are specialised when they need to be differentiated from existing ones. In comparison to corpora-based IC computation models, intrinsic IC computation models consider that abstract ontological concepts with many hyponyms are more likely to appear in a corpus because they can be implicitly referred in text by means of all their specialisations. In consequence, concepts located at a higher level in the taxonomy with many hyponyms or leaves (i.e. specialisations) under their taxonomic branches would have less IC than highly specialised concepts (with many hypernyms or subsumers) located on the leaves of the hierarchy.

In a recent work [21], intrinsic IC calculus is improved by incorporating into the assessment additional semantic evidences extracted from the background ontology. The $p(c)$ is estimated as the ratio between the number of leaves in the taxonomical hierarchy under the concept c (as a measure of c's generality) and the number of taxonomical subsumers above c including itself (as a measure of c's concreteness) (4). It is important to note that in case of multiple inheritance all the ancestors are considered. Formally:

$$IC(c) = -\log p(c) \cong -\log\left(\frac{\frac{|leaves(c)|}{|subsumers(c)|}+1}{max\_leaves+1}\right) \tag{4}$$

The above ratio has been normalised by the least informative concept (i.e. the root of the taxonomy), for which the number of leaves is the total amount of leaves in the taxonomy (*max_leaves*) and the number of subsumers including itself is 1. To produce values in the range 0..1 (i.e. in the same range as the original probability) and avoid *log*(0) values, 1 is added to the numerator and denominator.

As discussed in [20, 21] this approach represents an improvement to previous ones [19, 24] in that it can differentiate concepts with the same number of hyponyms/leaves but different degrees of concreteness (expressed by the number of subsumers that normalises the numerator). Moreover, it can also consider the additional knowledge modelled by means of multiple inheritance relationships. Finally, it prevents the dependence on the granularity and detail of the inner taxonomical structure by relying on taxonomic *leaves* rather than complete sets of hyponyms.

Intrinsic IC-based approaches overcome most of the problems observed for corpus-based IC approaches (specifically, the need of corpus processing and data-sparseness). Moreover, they achieve a similar, or even better accuracy than corpus-based IC calculation when applied over detailed and fine grained ontologies [21]. However, for small or very specialised ontologies with a limited taxonomical depth and low branching factor, the resulting IC values could be too homogenous to enable a proper differentiation of concepts' meanings [21].

2.2 Semantic Similarity from Multiple Ontologies

The main drawback of all the above ontology-based similarity measures is their complete dependency on the degree of coverage and detail of the input ontology [5]. Hence, ontology-based measures require a unique, rich and consistent knowledge source with a relatively homogeneous distribution of semantic links and good inter-domain coverage to work properly [23]. However, this is sometimes hard to achieve since, in many domains, knowledge is spread through different ontologies.

To tackle this problem, some authors [13, 26] focused on exploiting multiple ontologies for similarity assessment. The use of multiple ontologies provides additional knowledge that helps to improve the similarity estimation and to solve cases in which terms are not contained in a unique ontology [26].

Semantic similarity assessment from multiple ontologies is challenging because different ontologies may present significant differences in their levels of detail, granularity and semantic structures, making the comparison and integration of similarities computed from such different sources difficult [26]. In some approaches [13, 14], the two ontologies are simply connected by creating a new node which is a direct subsumer of their roots. This avoids the problem of knowledge integration but poorly captures possible commonalities between ontologies. Other authors [26] base their proposal in the differentiation between *primary* and *secondary* ontologies, so that secondary ontologies are connected to concepts with the same textual label in the primary ontology. Since such work relies on absolute path lengths to compute similarity (which depend on the size of each ontology), the authors scale path lengths taking as reference the size of the primary ontology.

In any case, the core task in multi-ontology similarity assessment is the discovery of equivalent concepts between the different ontologies, which can be used as bridges between the ontologies and thus, enabling a standard similarity calculus from the integrated structure [13, 26–28]. In the following we detail recent works proposed by the authors on multi-ontology similarity assessment that propose solutions for different similarity calculus paradigms.

2.2.1 A Multi-Ontology Semantic Similarity Method Based on Ontological Features

The method presented in [28] considers all the possible situations according to which ontology the compared concepts belong, and computes similarities according to the feature-based approach formalised in Eq. (1). Three cases are distinguished:

- *Case* 1: If the pair of concepts occurs in a unique ontology, the similarity is computed like in a mono-ontology setting (using e.g. Eq. (1)).
- *Case* 2: If the two concepts appear at the same time in several ontologies, each one modelling knowledge in a different but overlapping way, the similarity calculus will depend on the different levels of detail or knowledge representation accuracy of each ontology [13]. Given the nature of the ontology engineering process,

and the psychological implications of a human assessment of the similarity, two premises can be derived. First, ontological knowledge modelling is the result of a manual and explicit engineering process performed by domain experts. However, because of the bottleneck that characterises manual knowledge modelling, ontological knowledge is usually partial and incomplete [29]. As a result, if two concepts appear to be semantically distant, one cannot ensure if this is an implicit indication of semantic disjunction between the compared concepts or the result of partial or incomplete knowledge modelling. Second, as demonstrated in psychological studies [30], humans pay more attention to common than to differential features of the compared entities. As a conclusion of the two previous premises, given a pair of concepts appearing in different ontologies, the method in [28] considers the highest similarity score as the most reliable estimation and, thus, computes the similarity as follows:

$$sim(c_1, c_2) = \max_{\forall O_i \in O | c_1, c_2 \in O_i} sim_{O_i}(c_1, c_2) \tag{5}$$

- *Case* 3: If each of the two concepts belongs to a different ontology, each one modelling the knowledge from a different point of view, the set of shared and non-shared subsumers from the ontologies are assessed as follows: the set of shared subsumers for c_1 belonging to the ontology o_1 and c_2 belonging to the ontology o_2, $(T_{O1}(c_1) \cap T_{O2}(c_2))$ is composed by those subsumers of c_1, and c_2 with the same label, and also the set of ancestors of these terminological equivalent subsumers.

$$T_{O_1}(c_i) \cap T_{O_2}(c_i) = \bigcup_{\forall c_i \in ES} (T_{O_1}(c_i) \cup T_{O_2}(c_i)) \tag{6}$$

where $T_{O1}(c_1)$ and $T_{O2}(c_2)$ are defined as the set of subsumers of the concept χ_1 belonging to the ontology o_1, including the concept χ_1 , and the set of subsumers of the concept c_2 belonging to the ontology o_2, including the concept c_2. The set of terminologically equivalent superconcepts (*ES*) is defined as:

$$ES = \{ c_i \in T_{O_1}(c_1) \mid \exists c_j \in T_{O_2}(c_2) \wedge c_i \equiv c_j \} \tag{7}$$

Notice that "≡" means a terminological match (i.e. identical concept labels). The remaining elements in $T_{O1}(c_1) \cup T_{O2}(c_2)$ are considered as non-common subsumers. Once the set of common and non-common subsumers has been defined, the similarity measure defined in Eq. (1) can be directly applied.

2.2.2 A Multi-Ontology Semantic Similarity Method Based on IC

On the contrary to feature-based similarities, as stated above, IC-based measures rely on the ability to discover an appropriate Least Common Subsumer (LCS) that subsumes the meaning of the concepts belonging to different ontologies, and the

ability to coherently integrate IC values computed from different ontologies. In [31] a multi-ontology semantic similarity method based on IC is presented, which also considers the three cases detailed above:

- *Case* 1: Both concepts appear in a unique ontology, so that the LCS can be retrieved unequivocally from it and the similarity can be computed in the same manner as in a mono-ontology scenario.
- *Case* 2: If both concepts appear in several ontologies at the same time, several LCSs can be retrieved. In this case, it is necessary to decide which LCS is the most suitable to compute inter-concept similarity. Following the same premises derived for the previous method, the *most specific* LCS from those retrieved from the overlapping set of ontologies is considered. In terms of IC, the *most specific* LCS corresponds to the LCS candidate with the maximum IC value:

$$LCS(c_1, c_2) = \underset{\forall o \in O | c_1, c_2 \in o}{\arg\max} \; (IC(LCS_o(c_1, c_2)) \tag{8}$$

  where $LCS_o(c_1, c_2)$ is the *LCS* between c_1 and c_2 for the ontology $o \in O$.
- *Case* 3: If each concept belongs to a different ontology, the set of subsumers of each concept is retrieved. Then, both sets are compared to find *equivalent* subsumers (i.e. those with the same textual label). As a result, the two ontologies can be connected by a set of equivalent concepts and the LCS for the concept pair can be retrieved as the *least common equivalent subsumer*:

$$LCS(c_1, c_2) = Least\_Common\_Equivalent\_Subsumer(c_1, c_2) \tag{9}$$

where the *Least_Common_Equivalent_Subsumer* is a function that terminologically compares all the subsumers of c_1 in o_1 and c_2 in o_2 , and selects the most specific common one. In any case, the IC value of the retrieved LCS will be different when computing it from an ontology than from the other. In this case the maximum IC value is selected.

$$IC(LCS(c_1, c_2)) = \max_{o \in \{o_1, o_2\}} IC_o(LCS(c_1, c_2)) \tag{10}$$

2.2.3 A Method to Discover Semantically Equivalent LCS Between Different Ontologies

Finally, in [27] a method is proposed that complements the strict terminological matching of subsumers, in which the previous approaches rely, with a structural similarity function that aims at discovering semantically similar but not necessarily terminologically identical subsumers. This process is illustrated in Fig. 2.

Essentially, the method assesses the similarity of subsumer pairs according to their *semantic overlap* and their *structural similarity*. The former is computed according to the number of hyponyms that subsumer pairs have in common.

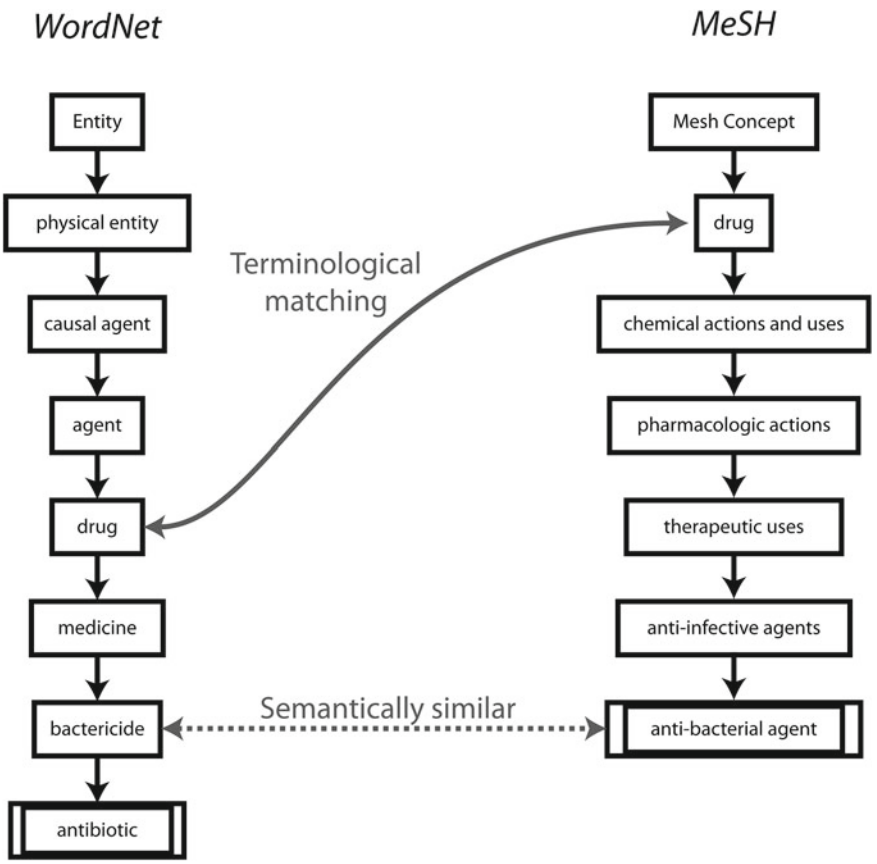


Fig. 2 Taxonomies for *antibiotic* (in the WordNet ontology) and *anti-bacterial agent* (in the MeSH ontology)

$$sem\_overlap(s_1, s_2) = \frac{|total\_hypo_{O1}(s_1) \cap total\_hypo_{O2}(s_2)|}{\sqrt{|total\_hypo_{O1}(s_1)| \times |total\_hypo_{O2}(s_2)|}} \quad (11)$$

where $total\_hypo_O(s_i)$ is the complete set of hyponyms of the subsumer s_i in the ontology O, and the intersection ($\cap$) between both sets of hyponyms is defined as the set of concepts that are terminologically equivalent. The motivation under this score is the fact that hyponyms of a concept summarise and bind its meaning and enable to differentiate it from other ones. Interpreting this principle in an inverse manner, the fact that two subsumers (each one modelled in a different ontology) share a certain amount of hyponyms, gives us an evidence of similarity.

Finally, *structural similarity* relies on the average *semantic overlap* of the immediate neighbourhood of the compared subsumers. The idea is to assess the similarity between two subsumers according to the semantic overlap between their pairwise

set of immediate ancestors and specialisations. Once all of the possible pairs of subsumers are evaluated, the pair with the highest structural similarity is selected as the LCS. In this manner, concepts with non-strictly identical labels but similar meanings could be matched, thus enabling a more accurate assessment of semantic similarity in a multi-ontology setting.

2.3 Distributional Similarities

Distributional approaches for similarity calculus only use textual corpora to infer concept semantics. They are based on the assumption that words with similar distributions have similar meanings [32]. Thus, they assess term similarities according to their co-occurrence in corpora. Because words may co-occur due to different kinds of relationships (i.e. taxonomic and non-taxonomic), distributional measures capture the more general notion of semantic *relatedness* in contrast to *similarity*, which is understood strictly as taxonomic resemblance. Distributional approaches can be classified into first order and second order co-occurrence measures.

2.3.1 First Order Co-occurrence Measures

They assume that related terms have a tendency to co-occur, and measure their relatedness directly from their explicit co-occurrence [33–35].

These measures usually rely on large raw corpora like the Web, from which term co-occurrence and, thus, similarity are estimated from the page-count provided by a Web Search Engine when querying both terms. Compared to IC-based measures relying on tagged corpora, distributional measures require neither any knowledge source nor manual annotation to support the assessment. Thanks to the coverage offered by a corpus as large and heterogeneous as the Web, distributional measures can be applied to terms that are not typically considered in ontologies such as named entities or recently minted terms.

However, their reliance on raw textual corpora is also a drawback. First, term co-occurrences estimated by page-counts omit the type of semantic relationship that motivated the co-occurrence. Many words co-occur because they are taxonomically related, but also because they are antonyms or by pure chance. Thus, page counts give a rough estimation of statistical dependency. Second, page counts deal with words rather than concepts. Due to the ambiguity of language and the omission of word context analysis, polysemy and synonymy may negatively affect the estimation of similarity/relatedness. Finally, page counts may not necessarily be equal to word frequencies because queried words might appear several times in a Web resource. Due to these reasons, some authors have questioned the usefulness of page counts alone as a measure of relatedness [35]. Other authors [36] also questioned the effectiveness of relying just on first order co-occurrences because studies on large corpora gave examples of strongly associated words that never co-occur [35].

2.3.2 Second Order Co-occurrence Measures

They estimate relatedness as a function of the co-occurrence of words appearing in the context in which the terms to evaluate occur [37–39].

These measures were developed to tackle the situation in which closely related words do not necessarily co-occur [36]. In such approaches, two words are considered similar or related to the extent that their contexts are similar [40]. Some approaches build contexts of words from a corpus of textual documents or the Web [41]. However, some authors have criticised their usefulness to compute relatedness because, while semantic relatedness is inherently a relation on concepts, Web-based approaches measure a relation on words [42]. Indeed, big-enough sense-tagged corpora are needed to obtain reliable concept distributions from word senses. Moreover, commercial bias, spam and noise are problems that may affect distributional measures when using the Web as corpus.

To minimise these problems, some authors exploited concept descriptions (*glosses*) offered by structured thesaurus like WordNet [39]. They argue that words appearing in a *gloss* are likely to be more relevant for the concept's meaning than text drawn from a generic corpus and, in consequence, glosses may represent a more reliable context. The use of glosses instead of the Web as corpora, results in a significant improvement of accuracy [39]. However, those measures are hampered by their computational complexity, since the creation of context vectors in such a big dimensional space is difficult. Moreover, because they rely on WordNet glosses, those measures are hardly applicable to other ontologies in which glosses or textual descriptions are typically omitted [15].

3 Applications to Data Privacy

Within the current context of Information Societies, governments, public administration and organisations usually require the interchange or release of potentially sensitive information. Because of the confidential nature of such information, appropriate data protection measures should be taken by the responsible organisations in order to guarantee the fundamental right to privacy of individuals. To guarantee such privacy, data protection/sanitisation methods obfuscate or remove sensitive information. This process necessarily incurs some loss of information, which may hamper data utility. Because the protected data should still be useful for third parties, data protection should balance the trade-off between privacy and data utility [1].

Traditionally, data protection has been performed manually, in a process by which a set of human experts rely on their knowledge to detect and minimise the risks of disclosure [43]. Semantics are in fact crucial in data privacy because they define the way by which humans (sanitisers, data analysts and also potential attackers) understand and manage information. Semantics thus directly influence the practical disclosure and also the analytical utility of the protected data, because they provide the *meaning* of data [2, 44]. However, manual protection efforts can hardly cope with the current needs of privacy-preserving information disclosing [45].

In recent years, a plethora of automated protection mechanisms have been proposed [46]. However, most of them manage data from a pure statistical/distributional perspective and neglect the importance of semantics [2, 3]. Moreover, because of their numerically-oriented design, current solutions mainly focus on structured databases and/or their implementations mostly assume univalued and numerical attributes [1], which can be easily evaluated, compared and transformed by using standard arithmetical operators. However, a large amount of the sensitive data that is involved nowadays in data releases (i.e. Open Data [47]) is of unstructured and textual nature. For such data, standard protection methods are either non-applicable, or neglect data semantics. The limitations of current works with regard to data semantics hamper both the practical privacy and the analytical utility of the protected outputs.

In this section, we discuss some recent works on data privacy that, by relying on the theory of semantic similarity and exploiting structured knowledge bases, aim at overcoming the limitations of pure statistical and distributional methods when dealing with textual data.

3.1 Design of Semantic Operators

Privacy preserving methods usually transform input data to make it less detailed or more homogenous, so that the chance of disclosing sensitive information is minimised. Typical mechanisms employed in the protection of structured databases (i.e. microdata) are:

- Data microaggregation, which consists on making groups of similar individuals (represented by records in a data base) and replacing them by a prototypical value, thus making them indistinguishable from the other members of the group.
- Noise addition, which adds a degree of uncertainty to the input data proportional to the range of such data, in order to lower the probability of an unequivocal disclose of sensitive information.
- Data recoding, which replace individual values by, usually, generalised versions in order to make them indistinguishable.
- Sampling, which selects a subset of individuals as representatives for the whole dataset.
- Data swapping, which reciprocally replaces attribute values of similar individuals (records) in order to avoid unequivocal disclosure.

Most of the above mechanisms require a set of basic operators in order to compare and transform input data. Specifically:

- A distance measure is needed to detect which individuals are most similar in order to group/microaggregate/swap their values and minimise the loss of information resulting from the subsequent data transformation.
- An averaging operator is usually needed to select a prototypical value as the central point of a sample or to sort such sample according to their distance to the most central value.

- The variance of a sample of values is relevant when measuring, for example, the magnitude of the noise to be added to distort input values.

When dealing with numerical data, the standard arithmetical operators used to measure distances, compute arithmetic averages, standard deviations or variances can be straightforwardly applied. However, when managing textual data, such operators do not make sense. To tackle this problem, in [44, 48, 49], the authors propose several operators analogous to arithmetical ones that can manage textual data and that are both semantically and mathematically coherent. Essentially, they rely on the notion of semantic similarity to compute the resemblance between textual term pairs. By means of these operators one is able:

- To select the average value of a sample or the centroid as the value that minimises the semantic distance to all other values of the sample.
- To sort data according to their distance to the least central value (i.e. the most marginal one [49]).
- To measure the variance of a sample as the standard variance between their aggregated semantic distances.

Empirical experiments conducted in [44, 48, 49] show how the proposed operators are able to better capture the meaning of input textual data than pure distributional methods and, thus, to better preserve the utility of the output when applied to privacy-protection methods. Moreover, in [49], the proposed operators are proven to be coherent from a mathematical perspective, which is relevant when those are applied to existing algorithms designed for numerical data (e.g. hierarchical clustering [50]).

3.2 Adaptation of Privacy Protection Mechanisms to Textual Attributes

Once a set of semantically and mathematically coherent operators is available, they can be used to adapt existing protection mechanisms so that the semantics of textual data are also considered.

Within the context of structured databases (e.g. census databases), in [3, 49, 51], the authors rely on the basic semantic similarity calculus and on the ability to compute the centroid of a sample of textual values in order to apply a well-known microaggregation algorithm (MDAV: Maximum Distance to the Average Vector [52]) to textual data in a semantically-coherent way. Specifically, in [49], it is shown how such a process is trivial if the appropriate operators are available and in [3] it is shown how the inherent characteristics of textual data (which are discrete and usually take values from a finite set of categories) can be considered in the algorithm design to better retain the utility of the protected output.

In [53], it is shown how a resampling algorithm meant for numerical data can also deal with textual values by relying on semantically coherent sorting and averaging operators. Even though data resampling usually incurs in a higher information loss than microaggregation it is also significantly more efficient, as it is empirically shown in [44], which may be preferable when protecting large data sets.

Finally, in [54] a new recoding method based on iteratively replacing semantically similar values is proposed. Several heuristics designed to preserve data semantics as much as possible and make the process more efficient are also defined.

In [44], an empirical study of the three above-described mechanisms is performed, comparing them accordingly to the dimensions of utility preservation, disclosure risk and computational efficiency, respectively.

All the above mechanisms focus on making input data more homogenous and indistinguishable in order to fulfil the *k*-anonymity privacy guarantee [55] over structured databases, that is, the fact that each individual (record) contained in a data set (database) is indistinguishable from, at least, k−1 other individuals; thus, the practical probability of re-identification of an individual is reduced to 1/k.

There exist other privacy models which offer different and more robust privacy guarantees. The most well-known is the ε-differential privacy model [56], which requires the protected data to be insensitive to modifications of *one* input record with a probability depending on ε. In this manner, an attacker would not be able to unequivocally disclose the confidential information of a specific individual with a probability depending on ε. To achieve this guarantee, practical enforcements focusing on numerical data add noise to the attribute values in a magnitude so that the protected output becomes insensitive (according to the ε parameter) to a modification of an input record.

In [57, 58], the authors propose mechanisms to achieve the ε-differential privacy guarantee for textual attributes in structured databases. The proposed method relies on a modified version of the MDAV algorithm that is applied to microaggregate the input values to reduce the sensitivity of data. Then, instead of adding numerical noise (which does not make sense for textual values), an exponential mechanism is used to replace the microaggregated values by the prototypes of each group (i.e. centroids) in a probabilistic way. The probability calculus picks the most probable centroid according to its degree of semantic centrality towards the other elements of the group, which is computed as detailed above [49]. In this manner, the degree of uncertainty and, thus, of information loss (from a semantic perspective) of differentially-private outputs is significantly reduced in comparison with the standard mechanism based on Laplacian noise [56].

3.3 Protection of Semi-structured Data

The above mechanisms and privacy models focus on relational data bases, in which individuals are represented by means of records with a finite set of univalued attributes. In such cases, it is quite straightforward to compare record pairs, since they have the same set of attributes and cardinality.

There exist, however, data sets containing transactional data in which individuals' information is represented by lists of items with variable and usually high cardinality (e.g. lists of diagnoses, query logs of users of a web search engine, etc.). Moreover, such data sets usually contain textual values. The protection of such data sets has

been usually performed by generalising values to a common abstraction [59]. This process, however, severely hampers data utility due to the loss of granularity of the generalisation process, especially when generalising outlying values.

To tackle this issue, in [51, 60] the authors adapt the MDAVmicroaggregation method, which was originally designed for univalued numerical records, to achieve *k*-anonymity in transactional data sets with textual values. To do so, the authors first propose different mechanisms to aggregate the semantic similarities of sets of values with different cardinalities, so that the MDAV algorithm can be applied like in the univalued scenario. After that, an especially designed aggregation methodology is proposed so that the prototypes of each microaggregated group capture both the semantics and the distributional features of the set of transaction lists that they are aggregating. The evaluation performed over a real set of user query logs shows that the use of semantic similarity measures results in a better preservation of data utility than purely distributional approaches.

3.4 Sanitisation of Unstructured Text

In the area of document redacting and sanitisation, input data consists on unstructured textual documents (e.g. medical visit outcomes, e-mails) whose contents must be protected according to the kind of information that should not be revealed (e.g. sensitive diseases, religious orientations, addresses, etc.). The challenge here is that no a priori sets of quasi-identifiers can be defined because, potentially, any combination of words of any cardinality may disclosure sensitive information. Thus, two tasks are usually performed: (i) detection of terms that, individually (e.g. proper nouns) or, due to their co-occurrence (e.g. treatments and symptoms of sensitive diseases), may disclose sensitive data, and (ii) protection of such terms, either by simple removal (redaction) or generalisation (i.e. sanitisation). The latter is more desirable because it better preserves the semantics of the original document and, thus, its analytical utility.

Within this area, in [61] an automatic method is proposed to detect sensitive terms individually, by using their degree of informativeness to measure the amount of sensitive semantics that they disclose. Then, these sensitive terms are replaced by generalisations obtained from a knowledge base. In [62] this method is extended in order to detect sets of terms that, because of their co-occurrence, may disclosure more sensitive information via semantic inference. In this latter work, first-order distributional similarity measures computed from the Web (as corpora) are used to quantify the degree ofdisclosure that combinations of terms enable with regard to a sensitive one (e.g. combinations of treatments and symptoms with respect to a sensitive disease). This is done as a function of their semantic relatedness that, as stated in the previous chapter, is captured by the distributional measures.

In [63, 64] the authors also focus on the anonymisation of textual documents. They rely on document vector spaces, which are normally used in information retrieval systems, to characterise a document as a vector of terms with frequency-based weights. Then, such vectors are microaggregated under the *k*-anonymity model using their cosine-distance as similarity measure.

3.5 Evaluating the Semantic Data Utility

In general, the data utility of anonymised data is retained if the same conclusions can be extracted from the analysis of the original and the protected data sets. When dealing with textual data, such utility should be understood from a semantic perspective [2, 3].

In [51, 54], the authors propose a semantically-grounded method toevaluate the *information loss* (and thus, the degree of utility preservation) of privacy-protected outputs. They rely on semantic clustering algorithms [65] able manage textual data and build clusters according to the semantic similarity between textual entities. To measure the information loss resulting from the data transformation performed during the anonymisation process, the authors quantify the differences between the cluster set obtained from original data against that obtained from the masked version. Because such cluster sets are a function of data semantics, their resemblance is a function of the preservation of data semantics during the protection process and, thus, of data utility (i.e. similar cluster sets indicate that similar analytical conclusions can be extracted from both the original and masked datasets).

In [63] the semantic data utility is evaluated from an information retrieval perspective, by performing specific "utility queries" and computing standard metrics of precision and recall.

3.6 Measuring the Semantic Disclosure Risk

The practical privacy achieved by a protection method is measured as the risk of re-identification of the original records. This is usually quantified as the percentage of records in the original data set that can be correctly matched from the protected output, that is, the percentage of Record Linkages (RL).

When dealing with numerical data, RL is usually measured according to the number of correct matches between the records whose values are the least distant. In order to apply the same process when protecting textual data sets, the notion of semantic similarity can be used instead of the usual arithmetical distance. In [66], the authors propose a semantically-grounded RL method that quantifies the similarity between the semantics of record values, and perform an analysis according to the kind of features of the knowledge structure exploited to guide that assessment. In this work, the use of semantic similarities to guide the linkage process produced a higher number of correct linkages than the standard non-semantic approach (which is just based on comparing value labels), thus providing a more realistic evaluation of disclosure risk.

In [44, 51], the semantic RL method was also used to evaluate the practical privacy of the semantically-grounded protection mechanisms discussed above.

In [63] risk evaluation was considered from the information retrieval perspective by formulating queries containing risky terms.

4 Conclusions

In this chapter we have classified and discussed recent works proposedby the authors in the area of semantic similarity. Available solutions offer a wide spectrum of tools and methods to cover the different needs (e.g. accuracy, efficiency, etc.) and availability of external resources (e.g. ontologies, tagged or raw corpora) of specific application scenarios.

A relevant application scenario of semantic similarity is the protection of textual data. Even though privacy protection algorithms have been traditionally designed for numerical data, in recent years, there is a growing interest in applying them to textual data. Semantic similarity has a crucial role in such scenarios, because it is the key to enable the semantically coherent comparisons and transformations of data that privacy-protection algorithms require. This chapter summarised recent approaches in this direction, which either adapt well-known protection algorithms to structured textual data sets (i.e. tabular data) and/or add support for less structure textual data, such as transactional data sets or unstructured textual documents. Data semantics has been also considered during the evaluation of the protected outputs. Empirical experiments reported in those work shave shown that by carefully considering semantics during the data protection process, the analytical utility of the protected outputs can be better preserved in comparison with purely statistical or distributional approaches.

Acknowledgments Authors are solely responsible for the views expressed in this chapter, which do not necessarily reflect the position of UNESCO nor commit that organisation. 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, ICWT TIN2012-32757, ARES-CONSOLIDER INGENIO 2010 CSD2007-00004, CO-PRIVACY TIN2011-27076-C03-01 and BallotNext IPT-2012-0603-430000) and by the Government of Catalonia (under grant 2009 SGR 1135).

References

1. Domingo-Ferrer, J.: A survey of inference control methods for privacy-preserving data mining. In: Aggarwal, C.C., Yu, P.S. (eds.) Privacy-Preserving Data Mining, pp. 53–80. Springer, Berlin (2008)
2. Torra, V.: Towards knowledge intensive data privacy. In: Proceedings of the 5th International Workshop on Data Privacy Management, pp. 1–7. Springer, Berlin (2011)
3. Martínez, S., Sánchez, D., Valls, A.: Semantic adaptive microaggregation of categorical microdata. Comput. Secur. **31**, 653–672 (2012)
4. Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., Swartout, W.R.: Enabling technology for knowledge sharing. AI Mag. **12**, 36–56 (1991)
5. Cimiano, P.: Ontology Learning and Population from Text: Algorithms. Evaluation and Applications. Springer, Berlin (2006)
6. Stumme, G., Ehrig, M., Handschuh, S., Hotho, S., Madche, A., Motik, B., Oberle, D., Schmitz, C., Staab, S., Stojanovic, L., Stojanovic, N., Studer, R., Sure, Y., Volz, R., Zacharia, V.: The karlsruhe view on ontologies. University of Karlsruhe, Institute AIFB, Germany, Technical report (2003)

7. Rada, R., Mili, H., Bichnell, E., Blettner, M.: Development and application of a metric on semantic nets. IEEE Trans. Syst. Man Cybern. **9**, 17–30 (1989)
8. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: 32nd Annual Meeting of the Association for Computational Linguistics, pp. 133–138. Association for Computational Linguistics (1994)
9. Leacock, C., Chodorow, M.: Combining Local Context and WordNet Similarity for Word Sense Identification. WordNet: An Electronic Lexical Database, pp. 265–283. MIT Press, Cambridge (1998)
10. Li, Y., Bandar, Z., McLean, D.: An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans. Knowl. Data Eng. **15**, 871–882 (2003)
11. Batet, M., Sánchez, D., Valls, A.: An ontology-based measure to compute semantic similarity in biomedicine. J. Biomed. Inf. **44**, 118–125 (2011)
12. Sánchez, D., Batet, M., Isern, D., Valls, A.: Ontology-based semantic similarity: a new feature-based approach. Expert Syst. Appl. **39**, 7718–7728 (2012)
13. Rodríguez, M.A., Egenhofer, M.J.: Determining semantic similarity among entity classes from different ontologies. IEEE Trans. Knowl. Data Eng. **15**, 442–456 (2003)
14. Petrakis, E.G.M., Varelas, G., Hliaoutakis, A., Raftopoulou, P.: X-similarity:computing semantic similarity between concepts from different ontologies. J. Digital Inf. Manage. **4**, 233–237 (2006)
15. Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V., Sachs, J.: Swoogle: A search and metadata engine for the semantic web. In: Thirteenth ACM International Conference on Information and Knowledge Management, CIKM 2004, pp. 652–659. ACM Press, New York (2004)
16. Resnik, P.: Using information content to evalutate semantic similarity in a taxonomy. In: 14th International Joint Conference on Artificial Intelligence, IJCAI 1995, pp. 448–453. Morgan Kaufmann Publishers Inc., Burlington (1995)
17. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: International Conference on Research in Computational Linguistics, ROCLING X, pp. 19–33 (1997)
18. Lin, D.: An Information-theoretic definition of similarity. In: Fifteenth International Conference on Machine Learning, ICML 1998, pp. 296–304. Morgan Kaufmann, Burlington (1998)
19. Seco, N., Veale, T., Hayes, J.: An intrinsic information content metric for semantic similarity in wordNet. In: 16th European Conference on Artificial Intelligence, ECAI 2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004, pp. 1089–1090. IOS Press, Valencia (2004)
20. Sánchez, D., Batet, M.: A new model to compute the information content of concepts from taxonomic knowledge. Int. J. Semant. Web Inf. Syst. **8**, 34–50 (2012)
21. Sánchez, D., Batet, M., Isern, D.: Ontology-based Information Content computation. Knowl. Based Syst. **24**, 297–303 (2011)
22. Sánchez, D., Batet, M., Valls, A., Gibert, K.: Ontology-driven web-based semantic similarity. J. Intell. Inf. Syst. **35**, 383–413 (2009)
23. Pirró, G.: A semantic similarity metric combining features and intrinsic information content. Data Knowl. Eng. **68**, 1289–1308 (2009)
24. Zhou, Z., Wang, Y., Gu, J.: A new model of information content for semantic similarity in wordNet. In: Second International Conference on Future Generation Communication and Networking Symposia, FGCNS 2008, pp. 85–89. IEEE Computer Society (2008)
25. Blank, A.: Words and concepts in time: towards diachronic cognitive onomasiology. In: Eckardt, R., von Heusinger, K., Schwarze, C. (eds.) Words and Concepts in Time: Towards Diachronic Cognitive Onomasiology, pp. 37–66. Mouton de Gruyter, Berlin, Germany (2003)
26. Al-Mubaid, H., Nguyen, H.A.: Measuring semantic similarity between biomedical concepts within multiple ontologies. IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. **39**, 389–398 (2009)
27. Sánchez, D., Solé-Ribalta, A., Batet, M., Serratosa, F.: Enabling semantic similarity estimation across multiple ontologies: an evaluation in the biomedical domain. J. Biomed. Inf. **45**, 141–155 (2012)

28. Batet, M., Sánchez, D., Valls, A., Gibert, K.: Semantic similarity estimation from multiple ontologies. Appl. Intell. **38**, 29–44 (2013)
29. Gómez-Pérez, A., Fernández-López, M., Corcho, O.: Ontological Engineering. Springer, Berlin (2004)
30. Tversky, A.: Features of similarity. Psycological Rev. **84**, 327–352 (1977)
31. Sánchez, D., Batet, M.: A semantic similarity method based on information content exploiting multiple ontologies. Expert Syst. Appl. **40**, 1393–1399 (2013)
32. Waltinger, U., Cramer, I., TonioWandmacher: from social networks to distributional properties: a comparative study on computing semantic relatedness. In: Thirty-First Annual Meeting of the Cognitive Science Society, CogSci 2009, pp. 3016–3021. Cognitive Science Society (2009)
33. Turney, P.D.: Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: 12th European Conference on Machine Learning, ECML 2001, pp. 491–502. Springer, Berlin (2001)
34. Cilibrasi, R.L., Vitányi, P.M.B.: The google similarity distance. IEEE Trans. Knowl. Data Eng. **19**, 370–383 (2006)
35. Bollegala, D., Matsuo, Y., Ishizuka, M.: A relational model of semantic similarity between words using automatically extracted lexical pattern clusters from the web. In: Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, pp. 803–812. ACL and AFNLP, (2009)
36. Lemaire, B., Denhière, G.: Effects of high-order co-occurrences on word semantic similarities. Current Psychol. Lett. Behav. Brain Cogn. **18**, 1 (2006)
37. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. In: 18th International Joint Conference on Artificial Intelligence, IJCAI 2003, pp. 805–810. Morgan Kaufmann, Burlington (2003)
38. Wan, S., Angryk, R.A.: Measuring semantic similarity using wordNet-based context Vectors. In: IEEE International Conference on Systems, Man and Cybernetics, SMC 2007, pp. 908–913. IEEE Computer Society (2007)
39. Patwardhan, S., Pedersen, T.: Using wordNet-based context vectors to estimate the semantic relatedness of concepts. In: EACL 2006 Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together, pp. 1–8 (2006)
40. Harris, Z.: Distributional structure. In: Katz, J.J. (ed.) The Philosophy of Linguistics, pp. 26–47. Oxford University Press, New York (1985)
41. Sahami, M., Heilman, T.D.: A Web-based kernel function for measuring the similarity of short text snippets. In: 15th International World Wide Web Conference, WWW 2006, pp. 377–386. ACM Press, New York (2006)
42. Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of semantic distance. Comput. Linguist. **32**, 13–47 (2006)
43. MRA Health Information Services, http://mrahis.com/blog/mra-thought-of-the-day-medical-record-redacting-a-burdensome-and-problematic-method-for-protecting-patient-privacy/
44. Martínez, S., Sánchez, D., Valls, A.: A semantic framework to protect the privacy of electronic health records with non-numerical attributes. J. Biomed. Inf. **46**, 294–303 (2013)
45. http://www.osti.gov/opennet
46. Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Nordholt, E.S., Spicer, K., Wolf, P.P.D.: Statistical Disclosure Control. Wiley, New York (2013)
47. Auer, S.R., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A Nucleus for a Web of Open Data. In: The Semantic Web, p. 722 (2007)
48. Martínez, S., Valls, A., Sánchez, D.: Semantically-grounded construction of centroids for datasets with textual attributes. Knowl. Based Syst. **35**, 160–172 (2012)
49. Domingo-Ferrer, J., Sánchez, D., Rufian-Torrel, G.: Anonymization of nominal data based on semantic marginality. Inf. Sci. **242**, 35–48 (2013)
50. Batet, M.: Ontology based semantic clustering. AI Commun. **24**, 291–292 (2011)
51. Batet, M., Erola, A., Sánchez, D., Castellà-Roca, J.: Utility preserving query log anonymization via semantic microaggregation. Inf. Sci. **242**, 49–63 (2013)
52. Domingo-Ferrer, J., Torra, V.: Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Min. Knowl. Dis. **11**, 195–212 (2005)

53. Martínez, S., Sánchez, D., Valls, A.: Towards k-anonymous non-numerical data via semantic resampling. In: Information Processing and Management of Uncertainty (IPMU), pp. 519–528 (2012)
54. Martínez, S., Sánchez, D., Valls, A., Batet, M.: Privacy protection of textual attributes through a semantic-based masking method. Inf. Fusion **13**, 304–314 (2012)
55. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. SRI International Report (1998)
56. Dwork, C.: Differential privacy. In: 33rd International Colloquium ICALP, pp. 1–12. Springer, Berlin (2006)
57. Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., Martínez, S.: Enhancing data utility in differential privacy via microaggregation-based k-anonymity. VLDB J. (2014) (in press)
58. Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., Martínez, S.: Improving the utility of differentially private data releases via k-anonymity. In: 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (2013)
59. Terrovitis, M., Mamoulis, N., Kalnis, P.: Privacy-preserving anonymization of set-valued data. In: VLDB Endowment, pp. 115–125 (2008)
60. Batet, M., Erola, A., Sánchez, D., Castellà-Roca, J.: Semantic anonymisation of set-valued data. In: 6th International Conference on Agents and Artificial Intelligence, pp. 102–112 (2014)
61. Sánchez, D., Batet, M., Viejo, A.: Automatic general-purpose sanitization of textual documents. IEEE Trans. Inf. Forensics Secur. **8**, 853–862 (2013)
62. Sánchez, D., Batet, M., Viejo, A.: Minimizing the disclosure risk of semantic correlations in document sanitization. Inf. Sci. **249**, 110–123 (2013)
63. Nettleton, D.G., Abril, D.: Document sanitization: measuring search engine information loss and risk of disclosure for the wikileaks cables. In: International Conference on Privacy in Statistical Databases, pp. 308–321 (2012)
64. Abril, D., Navarro-Arribas, G., Torra, V.: Towards a private vector space model for confidential documents. In: 28th Annual ACM Symposium on Applied Computing, pp. 944–945 (2013)
65. Batet, M.: Ontology-based semantic clustering. AI Commun. **24**, 291–292 (2011)
66. Martínez, S., Sánchez, D., Valls, A.: Evaluation of the disclosure risk of masking methods dealing with textual attributes. Int. J. Innovative Comput. Inf. Control **8**, 4869–4882 (2012)

An Information Retrieval Approach to Document Sanitization

David F. Nettleton and Daniel Abril

Abstract In this paper we use information retrieval metrics to evaluate the effect of a document sanitization process, measuring information loss and risk of disclosure. In order to sanitize the documents we have developed a semi-automatic anonymization process following the guidelines of Executive Order 13526 (2009) of the US Administration. It embodies two main and independent steps: (i) identifying and anonymizing specific person names and data, and (ii) concept generalization based on WordNet categories, in order to identify words categorized as classified. Finally, we manually revise the text from a contextual point of view to eliminate complete sentences, paragraphs and sections, where necessary. For empirical tests, we use a subset of the Wikileaks Cables, made up of documents relating to five key news items which were revealed by the cables.

1 Introduction

The 28th of November 2010 marks the occurrence of one of the largest releases of classified data online, when WikiLeaks, a non-profit organization, published more than 250,000 United States diplomatic cables that had been sent to U.S. international relations department between December 1966 and February 2010, by 274 of its consulates, embassies, and diplomatic missions around the world. From this large set of published documents there were over 115,000 labeled as "confidential" or "secret" and the remaining ones are unclassified by the official security criteria. According to the United States government the documents are classified at 4 levels: "Top

D.F. Nettleton
Universitat Pompeu Fabra, Barcelona, Spain
e-mail: david.nettleton@upf.edu

D.F. Nettleton · D. Abril (✉)
IIIA-CSIC Artificial Intelligence Research Institute - Spanish National Research Council, Barcelona, Spain
e-mail: dabril@iiia.csic.es

D. Abril
Universitat Autonòma de Barcelona, Barcelona, Spain

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_9

secret", "Secret", "Classified" and "Unclassified". These categories are assigned by evaluating the presence of information in a document whose unauthorized disclosure could reasonably be expected to cause identifiable or describable damage to the national security [1]. This type of information includes military plans, weapons systems, operations, intelligence activities, cryptology, foreign relations, storage of nuclear materials, and weapons of mass destruction. On the other hand, some of this information is often directly related to national and international events which affect millions of people in the world, who in a democracy may wish to know the decision making processes of their elected representatives, ensuring a transparent and open government. Therefore, releasing such amount of confidential data caused a great debate between those who uphold the freedom of information and those who defend the right to withhold information.

In the summer of 2010, WikiLeaks reached an agreement with some media partners from Europe and the United States to publish a set of cables in an edited form, removing the names of sources and other sensitive data. However, later on all the US Embassy cables [2] were published on the Internet fully unedited, in a "raw" state. That means that they included all kinds of confidential information such as emails, telephone numbers, names of individuals and certain topics, whose absence may not have significantly impaired the informative value of the documents with respect to what are now considered the most important revelations of the Cables.

The goal of this research is twofold. On the one hand, we have focused on new ways that could help to automate the concealment of confidential data. To do so, we have implemented a semi-automatic method to sanitize confidential unstructured documents, such as the released WikiLeaks documents. On the other hand, this research has also focused on finding new mechanisms to evaluate the information loss and the disclosure risk of a set of sanitized documents. We have proposed a technique relying on traditional information retrieval metrics which evaluates both the information loss and the risk of disclosure of a sanitized data set, by means of query comparisons.

This paper is organized as follows: the Sect. 2 briefly reviews the state of the art and related work which is followed by the Sect. 3 which presents the sanitization method. Then, in the Sect. 4 we describe the information loss and disclosure risk evaluation process. This is followed by the Sect. 5 which details the empirical results for information loss and risk of disclosure. Finally, in Sect. 6 we summarize the paper and detail future lines of work.

2 Related Work

Document sanitization is the process of declassification or reduction of a documents classification level, by means of removing the sensitive information from a document. Figure 1 is an example of a US government document that has been manually sanitized prior to release. In recent years there have been many efforts to automate or

Fig. 1 Sanitization example (*source* Wikipedia)

help people to perform the anonymization process by saving time and getting more accurate results.

Document sanitization consists of two main tasks. The first one is the detection of sensitive data within the text and once the sensitive information is spotted the second task is performed, that consists in hiding the previously detected information, with the aim of minimizing the disclosure risk, while causing the least distortion to the document content. The first task is usually solved by Named Entity Recognition and Classification systems, which are a set of techniques developed by a subfield of Information Retrieval that intends to identify and classify atomic elements and entities which appear within a text. The second task has been studied and carried out in several ways; below we briefly describe some of them.

Chakaravarthy et al. in [3] present the Efficient RedAction for Securing Entities (ERASE) system for the automatic sanitization of unstructured text documents. The system prevents disclosure of protected entities by removing certain terms from the document, which are selected in such a way that no protected entity can be inferred as being mentioned in the document by matching the remaining terms with the entity database. Each entity in the database is associated with a set of terms related to the entity; this set is defined as the context of the entity.

Saygin et al. [4] propose a sanitization approach that first automatically detects sensitive named entities, such as person and organization names, dates, credit card numbers, etc. and then those named entities are perturbed and generalized to hide the sensitive information, i.e., enforcing k-anonymity [5] at individual term level.

Cumby et al. in [6] present a privacy framework for protecting sensitive information in text data, while preserving known utility information. The authors consider the detection of a sensitive concept as a multiclass classification problem, inspired in feature selection techniques, and present several algorithms that allow varying levels of sanitization. They define a set D of documents, where each $d \in D$ can be associated with a sensitive category $s \in S$, and with a finite subset of non-sensitive utility categories $Ud \subset U$. They define a privacy level similar to k-anonymity [5], called k- confusability, in terms of the document classes.

Hong et al. in [7] present a heuristic data sanitization approach based on 'term frequency' and 'inverse document frequency' (commonly used in the text mining field to evaluate how relevant a word in a corpus is to a document). In [8], Samelin et al. present an redactable signature scheme (RSS) for ordered linear documents which allows for the separate redaction of content and structure. Chow et al. in [9] present a patent for a document sanitization method, which determines the privacy risk for a term by determining a confidence measure cs(*t1*) for a term *t1* in the modified version of the document relative to sensitive topics *s*. In the context of the sanitization of textual health data, [10] presents an automated de-identification system for free-text medical records, such as nursing notes, discharge summaries, X-ray reports, and so on.

Finally, Anandan et al. [11] focus on the protection of detected named entities by generalizing the sensitive words. This generalization relies on WordNet [12], an ontology that provides complete semantic relationship taxonomy between words. It groups English words into set of synonyms called *synsets*, and those groups are hierarchically organized, from the most general to most specific meaning. As this perturbation method relies on the semantic meaning of words it ensures less information loss in the sanitization process. Moreover, the authors present a measure, *t*-plausibility, to evaluate the quality of the sanitized documents from a privacy protection point of view. A generalized document holds the *t*-plausibility if at least *t* base documents can be generalized to a given sanitized document where a base document refers to one that has not been sanitized in any way.

3 Sanitization Method

In this section, a simple supervised sanitization method based on entity recognition and pattern-matching techniques is presented. The purpose of this method is to identify and delete all entities and sensitive terms within classified documents that could disclose confidential information. As shown in Fig. 2 we have divided the sanitization method in two steps. The first one performs the identification and anonymization of sensitive names or other personal information, while the second one performs the

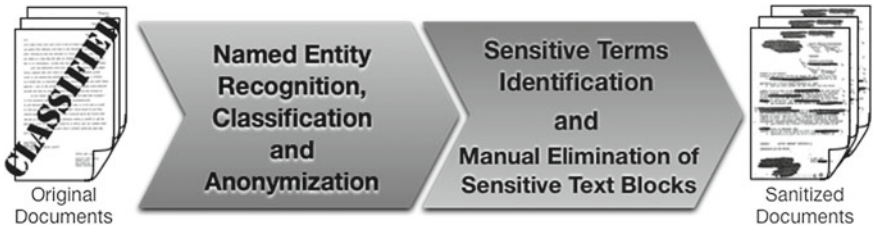


Fig. 2 Scheme for document sanitization

identification of text blocks which containing "risk" concepts, which later will be manually reviewed and eliminated. Both steps are described in detail below.

3.1 Step 1: Anonymization of Names and Personal Information of Individuals

To perform the first step we have used Pingar [13], an entity extraction software. This software identifies, classifies and anonymizes all named entities. It is able to detect the following named entities: people, organizations, addresses, emails, age, phone numbers, URLs, dates, times, money and amounts. The anonymization process is carried out replacing the identified sensitive information by its category plus an identification number. That is, {Pers1, Pers2, …}, {Loc1, Loc2, …}, {Date1, Date2, …} and so on. We also observe that the names of countries (Iran, United States, Russia, Italy, etc.) and places (London, Abu Dhabi, Guantanamo, etc) are unchanged in this process.

3.2 Step 2: Elimination of Text Blocks of "Risk Text"

This second step is divided in two sub-tasks; the identification of "risk" text blocks, which are those which contain the "risk" concepts, and the manual elimination of them. Unlike the first step, which hides/removes clear identifiers, such as personal information or locations, the goal of this second step, which is independent from the first step, is to detect and remove parts of the texts which contain risk terms. Due to the elimination of blocks of risk text, the main document information loss is incurred in this step.

The risk concepts are represented by 30 keywords extracted from Sect. 1.4 of Executive Order 13526 [1]. This section includes eight points (a) to (h) defining the topics that the US government considers of risk in terms of national security. In Table 1 there is the list of the first 30 initial risk terms. As a list of 30 concepts are not enough to figure out if a text makes reference to any of the stated points we have used the WordNet ontology database [12] to extend it. So, for each of these initial concepts we have extracted a set of new words related with its sense, i.e., synonyms and hyponyms. By hyponym we mean the lower part of the ontology tree starting from the given keyword, that is, more specific words. For example, "weapon" would give the following: "knife, sling, bow, arrow, rock, stick, missile, cannon, gun, bomb, gas, nuclear, biological, …". Finally we have obtained a list with a total of 655 risk terms (original + synonyms + hyponyms). We note that in this extraction process the word sense disambiguation was performed manually.

Then we processed the documents generating an output file in which all the keywords are signaled thus "****Keyword****", and which also indicates the relative

Table 1 Information retrieval metrics*

Metric	Formula	
Precision	$P = \frac{\lvert\{relevant\_docs\} \bigcap \{retrieved\_docs\}\rvert}{\lvert\{retrieved\_docs\}\rvert}$	(1)
Recall	$R = \frac{\lvert\{relevant\_docs\} \bigcap \{retrieved\_docs\}\rvert}{\lvert\{true\_relevant\_docs\}\rvert}$	(2)
F-measure	$F = 2.\frac{precision \cdot recall}{precision + recall}$	(3)
Coverage	$C = \frac{\lvert\{true\_relevant\_docs\_returned\}\rvert}{\lvert\{true\_relevant\_docs\}\rvert}$	(4)
Novelty	$N = \frac{\lvert\{false\_relevant\_docs\}\rvert}{\lvert\{total\_relevant\_docs\}\rvert + \lvert\{false\_relevant\_docs\}\rvert}$	(5)

*See [15] for more details of these metrics

distance of each "risk" keyword found from the start of the file. We cluster these distances for each file and use the information to signal documents with text areas that have a high density of risk keywords, which would be candidates to be eliminated from the file. We note that we applied a stemming process (using the Porter Stemming algorithm version 3 [14] implemented in Java, a process for reducing words to their stem/root form) to the keyword list and the words in the documents in order to match as many possible variants as possible of the root term. Finally, we manually revised the labeled files, using the clustered distance information for support, and deleted the paragraphs identified as having the highest clustering of "risk terms".

4 Information Loss and Risk Evaluation

In this section we present the method to evaluate the information loss and disclosure risk from a set of sanitized documents. This is performed by means of the results comparison when querying the original and the sanitized data set. In the Sect. 4.1 we describe the characteristics of the vectorial model search engine implemented and in 'Metrics' we define the information loss and risk metrics. We note that the same metrics are used to measure information loss and disclosure risk. However, these two metrics require different sets of queries (utility and risk queries) to perform the evaluation and give a different interpretation. The utility queries consist of terms about the general topic of each document set and the risk queries consist of terms that define sensitive concepts.

4.1 Search Engine

We have implemented our own search engine in Java, with the following main characteristics: an inverted index to store the relation between terms and documents and a hash-table to efficiently store the terms (vocabulary); elimination of stop-words and stemming; calculation of term frequency, inverted document frequency, root of

the sum of weights for the terms in each document; implementation of the Vectorial Model formula to calculate the relevance of a set of terms (query) with respect to the corpus of documents. Refer to [15] for a complete description of the Vectorial model and the formula used. We observe that the queries are by default 'OR'. That is, if we formulate the query "term1 term2 term3", as search engines do by default, an OR is made of the terms and the documents are returned which contain at least one of the three given terms, complying with "term1 OR term2 OR term3".

4.2 Information Loss and Risk of Disclosure Metrics

As a starting point, we have used a set of well-known information retrieval metrics, which are listed in Table 1 and briefly described below. The formulas are defined in terms of the following sets of documents: *true_relevant_documents* is the unchanged, non-sanitized, document set retrieved by the corresponding query by the Vectorial search engine. *Retrieved_documents* is the set returned by the search engine in reply to a given query that is above the relevance threshold, *relevant_documents*, are the documents above the relevance threshold which are members of the *true_relevant_documents* set. *True_relevant_docs_returned* are the documents in *true_relevant_documents* that are returned by the search engine in any position (above or below the threshold) and finally, *false_relevant_docs* are the documents not members of *true_relevant_documents* but which are returned above the relevance threshold. The degree of relevance of a document with respect to a query is calculated as a quantified value by the Vectorial model search engine, as commented in Sect. 4.1. We note that the assignment of the relevance thresholds is explained at the end of this section.

- *The Precision* is considered as the percentage of retrieved documents above the relevance threshold that are relevant to the informational query.
- The *Recall*, on the other hand, is considered as the percentage of retrieved documents above the relevance threshold that are defined as truly relevant.
- The *F-measure* (or balanced F-score) combines precision and recall and mathematically represents the harmonic mean of the two values.
- The *Coverage* is the proportion of relevant documents retrieved out of the total true relevant documents, documents known previously as being the correct document set for a given search.
- The *Novelty* is the proportion of documents retrieved and considered relevant which previously were not relevant for that query. That is, it measures the new information introduced for a given query. We interpret novelty as undesirable with respect to the quality of the results, because we assume that we have correctly identified the set of all true relevant documents.

As well as the four metrics listed in Table 1, we also consider four other measures:

- The average relevance of the documents whose relevance is above the relevance threshold.

- The total number of documents returned by the query whose relevance is greater than zero.
- The number of random documents which are members of the set of relevant documents for a given query.
- NMI (Normalized Mutual Information), we use an NMI type metric [16] for counting document's assignments to query document sets before and after sanitization.

That is, we compare the results of the document assignments to query sets by identifying the documents in each query document set before sanitization, and the documents which are in the same corresponding query document set after sanitization.

Quantification of information loss and risk: in order to obtain a single resulting value, we have studied all the parameters presented and defined a formula in terms of the factors which showed the highest correlation between the original and sanitized document metrics: F = F-measure, C = coverage, N = novelty, TR = total number of documents returned, PR = percentage of random documents in the relevant document set, and the NMI value. Hence IL, the information loss is calculated as:

$$\mathrm{IL} = \frac{(2 \times \mathrm{F}) + \mathrm{C} - \mathrm{N} + \mathrm{TR} - \mathrm{PR} - (2 \times \mathrm{NMI})}{8} \tag{6}$$

We observe that of the six terms in the formula, F and NMI are given a relative weight of 25 %, and the other four terms are given a relative weight of 12.5 %. The weighting was assigned by evaluating the relative correlations of the values before and after document sanitization for each factor. As the F-measure and the Normalized Mutual Information were the factors that showed the highest correlation between the original and sanitized document, we gave them a higher weight according to their correlation value with respect to the other values.

For the risk of disclosure, RD, we use the same formula and terms, however the interpretation is different: for IL a negative result represents a reduction in information, and for RD a negative result represents a reduction in risk.

Relevance threshold value for informational document sets. In order to apply the same criteria to all the search results, after studying the distributions in general of the relevance of the different queries, we chose a relevance of 0.0422 as the threshold. That is, we identify an inflexion point between the relevant documents (relevance greater or equal to 0.0422) and non-relevant documents (relevance less than 0.0422). See Table 2 as an example for the search results of a given query for which the first seven ranked documents (highlighted in grey) are above the relevance threshold.

Relevance threshold value for risk document sets. After studying the distributions of the relevance for each risk document set returned by the search engine, we assigned the relevance threshold of 0.010 for all the results sets, with the exception of result sets r9, r1 and r2 which were assigned a threshold of 0.020. The metric calculations then followed the same process as for the informational document sets.

Table 2 Example search results

Vector model search engine		
Search terms: query uq_{5-1}		
Query "putin berlusconi relations"		
Rank	Doc id	Relevance
1	u5.6	0.262488
2	u5.1	0.210500
3	u5.2	0.107093
4	u5.3	0.098520
5	u5.4	0.087844
6	u3.7	0.076260
7	u5.8	0.052028
8	u5.10	0.022432
…	….	……
44	ur.9	0.000034

5 Experimental Analysis

In this section we describe the documents set used and how we have obtained a set of classified documents. Then, we present the results for information loss and risk of disclosure, comparing query results between the original and the sanitized data set by means of the presented metrics.

5.1 Document Extraction

In order to test the proposed sanitization and evaluation techniques we have extracted a set of documents from the online Wikileaks Cable repository [2]. As in this online repository there are lots of documents related with different subjects, we selected the first five topics from the top ten revelations published by Yahoo! News [17]. We derived five queries corresponding to these five selected topics, as shown in Table 3. Then, we searched using these queries as keywords on www.cablegatesearch.net [2] to find the corresponding cables, thus obtaining a set of documents for each query. We observe that a sixth document set, i6, was randomly chosen from [2] for benchmarking purposes. The same five queries (Table 3) were used to test information loss (utility) in the empirical results section. Figure 3 shows a schematic representation of the process.

As was mentioned in the Sect. 3, we extracted 30 seed terms from the eight risk points defined in Section 1.4 of the US Executive Order 13526 [1], which are shown in Table 4. Hence, we defined eight different queries, one for each risk point, which are designated as $\{rq_1, \ldots, rq_8\}$, corresponding to document sets {r1, .., r8}. These

Table 3 Queries and documents used to test information loss

Id. query	Keywords (utility queries)	TC, CH[a]	ID[b]	Top five news item revelations (Yahoo!) [17]
uq_1	{Saudi, qatar, jordan, UAE, concern, iran, nuclear, program }	35, 10	il1	"Middle Eastern nations are more concerned about Iran's nuclear program than they've publicly admitted"
uq_2	{China, korea, reunify, business, united, states}	3, 3	il2	"U.S. ambassador to Seoul said that the right business deals might get China to acquiesce to a reunified Korea, if the newly unified power were allied with the United States"
uq_3	{Guantanamo, incentives, countries, detainees}	12, 10	il3	"The Obama administration offered incentives to try to get other countries to take Guantanamo detainees, as part of its plan to progressively close down the prison"
uq_4	{Diplomats, information, foreign, counterparts}	6, 6	il4	"Secretary of State Hillary Clinton ordered diplomats to assemble information on their foreign counterparts"
uq_{5-1}	{Putin, berlusconi, relations}	97, 10	il5	"Russian Premier Vladimir Putin and Italian Premier Silvio Berlusconi have more intimate relations than was previously known"
uq_{5-2}	{Russia, italy, relations}			
–	–	10, 10	il6[c]	–

[a] total cables, cables chosen
[b] informational document sets
[c] represents a set of randomly chosen documents to be used as a benchmark

terms were used in our sanitization processing to detect 'risk' text blocks, and were also employed to define eight different queries which are used to evaluate the risk.

We also defined a ninth query, rq_9, composed of all the terms from queries rq_1 to rq_8, whose corresponding document set is r9. This is included to act as a double check on the returned document set for all the terms together. Note that, due to how the vector model retrieves and ranks the documents, the document set returned by rq_9 may be different (in documents and their ordering) from the sum of the document sets returned by rq_1 to rq_8.

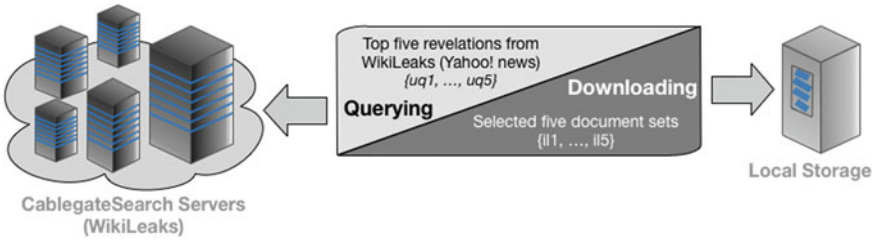


Fig. 3 Scheme for document extraction and querying

Table 4 Queries used to test risk of disclosure

Id. Query	Keywords (risk queries)	ID1	Classification categories, a→h, see [1]
rq_1	{Military, plan, weapon, systems}	r1	(a)
rq_2	{Intelligence, covert, action, sources}	r2	(b)
rq_3	{Cryptology, cryptogram, encrypt}	r3	(c)
rq_4	{Sources, confidential, foreign, relations, activity}	r4	(d)
rq_5	{Science, scientific, technology, economy, national, security}	r5	(e)
rq_6	{Safeguard, nuclear, material, facility}	r6	(f)
rq_7	{Protection, service, national, security}	r7	(g)
rq_8	{Develop, production, use, weapon, mass, destruction}	r8	(h)
rq_9	All terms from rq1 to rq8	r9	–

5.2 Information Loss

In Table 5 we see the results of applying the NMI metric to the original and sanitized document query sets. For the majority of query document sets, in general we see a relatively small information loss. In the case of query uq_{5-1}, the high information loss was due to the elimination of the named query terms, 'Putin' and 'Berlusconi', in the documents.

Table 5 Information Loss: percentage (%) differences of NMI metric for original and sanitized document corpuses (steps 1 + 2)

	uq_1	uq_2	uq_3	uq_4	uq_{5-1}	uq_{5-2}
Step 1	0.00	0.00	0.00	0.00	100.00	0.00
Step 2	11.00	0.00	14.00	50.00	100.00	0.00

Table 6 shows the percentage change for each metric value and informational document set, of the original documents and the sanitized documents processed by steps 1 and 2. The indicators used in the information loss formula (6) are highlighted in grey. The information loss calculated using formula (6) is shown in the rightmost column (IL), giving an average value of 26.1 %.

With reference to Table 6, we will now make some observations in terms of the percentage change for each metric and informational query, with the exception of query uq_{5-1}, as we have previously mentioned. We observe that the information loss is highest for query uq_4 (−38.62) and lowest for queries uq_1 and uq_3. If we look again at the terms which correspond to these queries (Table 3), those of queries uq_1 and uq_3 are more specific whereas those of query uq_4 are more general. Also, the correlation of the information terms of query uq_4 with the risk terms of all risk queries (Table 4) and/or their synonyms/hyponyms, is greater than that of queries uq_1 and uq_3. Another observation is the correlation of the different metrics with the information loss (IL). Again, excluding query uq_{5-1}, we see that PR, N and F correlate with the maximum values of IL (−14.37 and −38.62), whereas C is invariant; TR appears to be query dependant and has little correlation with the other metrics.

To summarize, Step 1 (*anonymization of names and personal information of individuals*) has little or no effect on the success of the informational queries, except those which contain specific names of people. This step preserves the confidentiality of the personal data of individuals who appear in the documents. Step 2 (*elimination of 'risk text'*) inevitably had a higher impact, given that blocks of text are eliminated from the documents. From the results of Table 6, we see that the information loss is query dependent, the PR, N and F indicators being the most consistent. By manual inspection of the documents, we can conclude in general that a worse value is due to the loss of key textual information relevant to the query. Also, queries with more general terms incur a higher information loss.

5.3 Disclosure Risk

We recall that the NMI metric measures the degree of correspondence between different groups. In Table 7 this metric is applied to the original and sanitized document query sets. A significant reduction can be seen in the correspondence, which contrasts with the results for the same metric applied to the information loss query document sets. Table 8 shows the percentage change for each of the metrics we described in

Table 6 Information loss: percentage (%) differences (**Δ**) of statistics for original and sanitized document corpuses (steps 1+2)

	Δ (P)	Δ (R)	Δ (F)	Δ (C)	Δ (N)	Δ (AR)	Δ (TR)	Δ (PR)	Δ (IL)
uq_1	−1.56	−12.50	−0.08	0.00	0.00	−38.15	−15.38	0.00	−6.625
uq_2	−40.00	0.00	−0.25	0.00	40.00	−0.38	−4.76	20.00	−14.37
uq_3	0.00	−14.29	−0.09	0.00	0.00	3.77	−12.50	0.00	−7.375
uq_4	−62.50	−75.00	−0.70	0.00	33.33	9.80	−10.81	25.00	−38.62
uq_{5-1}	−100.00	−100.00	−1.00	−100.00	−100.00	−100.00	−4.55	0.00	−75.62
uq_{5-2}	−11.11	0.00	−0.05	0.00	38.46	−5.03	0.00	0.00	−13.75

Legend P precision, *R* recall, *F* F measure, *C* coverage, *N* novelty, *AR* Average relevance for documents above threshold, *TR* total docs. returned, *PR* percentage of random docs in relevant doc set, *IL* percentage information loss calculated using formula (6)

Table 7 Risk of Disclosure: percentage (%) differences of NMI metric for original and sanitized document corpuses (steps 1 + 2)

rq_1	rq_2	rq_3	rq_4	rq_5	rq_6	rq_7	rq_8	rq_9
60.00	67.00	–	36.00	25.00	56.00	63.00	70.00	58.00

Table 8 Risk of Disclosure: percentage (%) differences (Δ) of statistics for original and sanitized document corpuses (steps 1 + 2)

	Δ (P)	Δ (R)	Δ (F)	Δ (C)	Δ (N)	Δ (AR)	Δ (TR)	Δ (PR)	Δ (RD)
rq_1	−66.67	−60.00	−0.64	−16.67	40.00	−26.94	−44.44	30.0	−47.37
rq_2	−66.67	−66.67	−0.67	−33.33	40.00	27.07	−48.39	16.7	−50.75
rq_3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	–
rq_4	−18.18	−35.71	−0.28	− 7.14	15.38	17.80	−4.17	1.96	−19.5
rq_5	−57.14	−25.00	−0.45	−12.50	50.00	11.74	−18.60	8.90	−28.87
rq_6	−60.00	−55.56	−0.58	−22.22	40.00	8.07	−55.26	17.8	−45.37
rq_7	−71.43	−50.00	−0.64	−12.50	55.56	−0.49	−33.33	35.7	−49.00
rq_8	−50.00	−70.00	−0.63	−50.00	23.08	−39.31	−29.41	23.3	−48.87
rq_9	−54.55	−58.33	−0.57	0.00	35.29	−14.29	−10.20	9.9	−35.62

Legend P precision, *R* recall, *F* F measure, *C* coverage, *N* novelty, *AR* Average relevance for documents above threshold, *TR* total docs. returned, *%PR* percentage of random docs in relevant doc set, *RD* percentage risk decrease calculated using formula (6)

Sect. 4.2, for each of the nine 'risk' queries, for the original documents and the sanitized documents of processing step 2. In general, we see a significantly greater percentage change in comparison to the information loss results of Table 6. The risk decrease calculated using formula (6) is shown in the rightmost column (RD), the average value being −47.26 %.

With reference to Table 8, we will now make some observations in terms of the percentage change for each metric and risk query. We observe that the risk reduction is greatest for queries rq_2, rq_7, rq_8, rq_1 and rq_6, with values of −50.75, −49.00, −48.87, −47.37 and −45.37, respectively. On the other hand, the risk reduction was least for queries rq_4 and rq_5, with values of −19.5 and −28.87, respectively. If we look again at the terms which correspond to these risk queries (Table 4), we see that those of queries rq_4 and rq_5 are more general, neutral terms, whereas those of the risk queries which corresponded to a greater risk reduction had more specific terms.

This was confirmed by the hyponyms and synonyms generated for each term by the WordNet API. This had a direct effect on identifying texts to be deleted given that the contexts in which the more specific terms are found tend to be (manually) identified as deletion candidates, whereas the generic terms may be used in a neutral context and are therefore not so often selected as deletion candidates.

Another observation is the correlation of the different metrics with the reduction in risk of disclosure (RD). We see that PR, TR, N and F correlate reasonably with

the highest values of RD. For example, the highest five values of RD coincide with the highest five values for PR and TR and F (although not in the same order).

6 Conclusions and Future Work

In this paper we have used information retrieval metrics to evaluate information loss and disclosure risk for a set of sanitized documents. In order to evaluate these two values we implemented a vectorial model search engine and also defined a formula to evaluate the information loss and disclosure risk by means of querying both document sets. The results show a relatively low overall information loss (16 % excluding query uq_{5-1}) for the utility queries (uq_1 to uq_5), whereas an average risk reduction of 47% was found for the risk queries (ur_1 to ur_9). As future work, we propose to improve the whole sanitization process; techniques such as term generalization can be developed for step 1 in order to reduce the information loss, as was proposed in [18]. Also, a greater automation of step 2 could be achieved by using semi-supervised learning methods applied to tagged examples. The aspect of automated risk term cluster analysis, although it is not the focus in the current work, could be considered as future work. Finally, we could consider using a learning process to find the best overall descriptive formula for information loss and disclosure risk.

Acknowledgments This research is partially supported by the Spanish MEC projects CONSOLIDER INGENIO 2010 CSD2007-00004 and eAEGIS TSI2007-65406-C03-02. The work contributed by the second author was carried out as part of the Computer Science Ph.D. program of the Universitat Autònoma de Barcelona (UAB).

References

1. Executive Order 13526, of the US Administration: Classified National Security Information, Section 1.4, points (a) to (h) (2009). http://www.whitehouse.gov/the-press-office/executive-order-classified-national-security-information
2. Wikileaks Cable repository. http://www.cablegatesearch.net
3. Chakaravarthy, V.T., Gupta, H., Roy, P., Mohania, M.K.: Efficient techniques for document sanitization. In: CIKM 2008, Napa Valley, California, USA, October 26–30 (2008)
4. Saygin, Y., Hakkani-Tr, D., Tur, G.: Sanitization and Anonymization of Document Repositories (2009)
5. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl. Based Syst. (IJUFKS) **10**(5), 557–570 (2002)
6. Cumby, C., Ghani, R.: A machine learning based system for semi-automatically redacting documents. In: Proceedings of IAAI 2011 (2011)
7. Hong, T.-P., Lin, C.-W., Yang, K.-T., Wang, S.-L.: A heuristic data-sanitization approach based on TF-IDF. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part I. LNCS, vol. 6703, pp. 156–164. Springer, Heidelberg (2011)
8. Samelin, K., Pöhls, H.C., Bilzhause, A., Posegga, J., de Meer, H.: Redactable signatures for independent removal of structure and content. In: Ryan, M.D., Smyth, B., Wang, G. (eds.) ISPEC 2012. LNCS, vol. 7232, pp. 17–33. Springer, Heidelberg (2012)

9. Chow, R., Staddon, J.N., Oberst, I.S.: Method and apparatus for facilitating document sanitization. US Patent Application Pub. No. US 2011/0107205 A1, May 5 (2011)
10. Neamatullah, I., Douglass, M.M., Lehman, L.H., Reisner, A., Villarroel, M., Long, W.J., Szolovits, P., Moody, G.B., Mark, R.G., Clifford, G.D.: Automated de-identification of free-text medical records. BMC Med. Inf. Decis. Making **8**, 32 (2008)
11. Anandan, B., Clifton, C., Jiang, W., Murugesan, M., Pastrana-Camacho, P., Si, L.: t-Plausibility: generalizing words to desensitize text. Trans. Data Priv. **5**(3), 505–534 (2012)
12. Miller, G.A., Beckwith, R., Fellbaum, C.D., Gross, D., Miller, K.: WordNet: an online lexical database. Int. J. Lexicograph **3**(4), 235–244 (1990)
13. Pingar: Entity extraction software. http://www.pingar.com
14. Porter, M.F.: An algorithm for suffix stripping. Program **14**(3), 130–137 (1980)
15. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval: The Concepts and Technology Behind Search, 2nd edn. ACM Press Books, England (2011)
16. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
17. Yahoo! News. Top 10 revelations from Wiki Leaks cables. http://news.yahoo.com/blogs/lookout/top-10-revelations-wikileaks-cables.html
18. Abril, D., Navarro-Arribas, G., Torra, V.: On the Declassification of Confidential Documents: Modeling Decision for Artificial Intelligence. Springer, Berlin (2011)

Part IV
Respondent Privacy: Location Privacy

Privacy for LBSs: On Using a Footprint Model to Face the Enemy

Mauro Conti, Roberto Di Pietro and Luciana Marconi

Abstract User privacy in Location Based Services (LBSs) is still in need of effective solutions. A new privacy model for LBSs has been recently proposed based on users' footprints—these being a representation of the amount of time a user spends in a given area. The model is claimed to be independent from the specific knowledge of the adversary about users' footprints. Despite this claim, we show in this chapter that when the adversary has a knowledge that differs from the one considered for the anonymization procedure, the model is not valid. Further, we generalize this weakness of the model and show that it is highly probable that the footprint model provides: (i) either a privacy level lower than the expected one; or, (ii) a LBS information coarser than what would be required for anonymization purposes. We support our claim via analysis: modeling the footprints data as an hypercube model; with a simple example to grasp the main problem; and, with the study of a real data set of traces of mobile users. Finally, we also investigate which properties must hold for both the anonymiser and the adversary knowledge, in order to guarantee an effective level of user privacy.

1 Introduction

The spreading of mobile devices, like smartphone, it is also spreading the usage of Location Based Services (LBSs). These services provide value to a user integrating her location with additional information. Hence, the identification of the user's geographical position is a constant characteristic of LBSs. However, the user might not

M. Conti
University of Padua, Padua, Italy
e-mail: conti@math.unipd.it

R. D. Pietro (✉)
University of Rome Tre, Rome, Italy
e-mail: dipietro@mat.uniroma3.it

L. Marconi
Sapienza University of Rome, Rome, Italy
e-mail: marconi@di.uniroma1.it

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_10

like to continuously disclose her location, not only to the network provider operation (that the user might trust to some extent), but also to other parties that provide the actual LBS. On the one hand, concerns about privacy might hinder a wide adoption of LBSs; on the other hand, users that are not really aware of leaking private information, might be disappointed.[1]

As further example of LBS, we can consider the vehicular services that many national transportation infrastructures are developing: traffic monitoring, hazard warning, congestion-based and 'pay-as-you-go' road pricing [8, 23]. It is easy to imagine the privacy threats the user of this system might be exposed to, e.g. the possibility to identify the user that requests a given service and her location at the time of the request. Providing privacy in these services is not an easy task. In fact, a user might be re-identified correlating the access information with other kind of information (e.g. the mobility of the user or some specific location-bound feature). In particular, there are three main issues related to the privacy of users in LBSs: (i) how to anonymize a user; (ii) how to specify the level of anonymity; (iii) how to guarantee to a given user the same level of desired anonymity for all of her requests. Common anonymization techniques leverage the concept of k-anonymity [7, 29] where: (i) consists of cloaking the user within a set of k potential users. The *feeling* based model, recently introduced [39, 40], also leverages the concept of k-anonymity. However, this model is motivated by the fact that specifying a practical value of k could be a difficult choice for the user. Hence, the feeling based model allows a user to define her desired level of anonymity (ii) by specifying a given area (e.g. a shopping mall). The entropy of the selected area is used to describe its popularity. In turns, the popularity is expressed in terms of footprints of the visitors in the selected area. The popularity of the user specified area is considered later on, in the subsequent user's LBSs requests, (iii) as the anonymization level that the LBS has to guarantee to the user. Among the many solutions and approaches proposed (see Sect. 2), we use the *feeling* based model as touchstone for our discussions. In fact, to the best of our knowledge, is the only proposal taking into account the "subjective" aspect of privacy. Any solution not considering that privacy is about people and not only about data, does not seem to have any practical application. While the feeling based approach seems to be promising from the point of view of user's awareness of privacy, we argue that the proposed solution do not take into account potential features the adversary could hold. In fact, the threat model considered in [39, 40] assumes an adversary having the same amount of information on the users as the one leveraged by the anonymiser. While this might seem a strong adversary model, it actually does not take into consideration practical aspects related to the distribution of such a knowledge over features like time, mode of transport, age or profession. We already proved this claim in [24, 25] considering one specific feature: time. In particular, in the cited work we considered both of the following situations to be practical. First, the adversary might have the information of the users footprints structured

[1] For instance, a German politician "discovered" his network operator collected 35,000 traces of his position in a period of 6 months. These data are now available to show the seriousness of the threat: http://www.zeit.de/datenschutz/malte-spitz-vorratsdaten.

over time (e.g. how many footprints in the morning and how many in the afternoon). Second, the adversary might just be able to observe a subset of the footprints (e.g. the adversary is only able to get footprints information during the morning). While in the former case the adversary is stronger than the one consider in [39, 40]—having more structure data—the latter scenario describes a weaker adversary—basing its decisions on depleted information. One could argue that a possible solution could be to extend the protocol in [39, 40] in order to handle time in a finer manner, so as to thwart the time-aware adversary. Actually, extending the analysis performed in [24, 25], we show that even considering the users spatio-temporal dimensions might still not be sufficient to guarantee the desired level of privacy. In fact, scenarios similar to the ones proposed for time, can be also obtained considering other realistic features like the user mode of transportation or even a combination of them.

The results of our generalization move the focus on the question that does really matter: the relationship between the anonymiser and the adversarial knowledge. Where the anonymiser and the adversary have not the same knowledge, privacy breaches might occur. As a consequence, the generalization also provides evidence of the hardness of protecting privacy for LBSs. In fact, on one hand we can deduce from our generalization that providing effective solutions for privacy-preserving LBS requires to tackle the assumption that the adversarial knowledge is unknown to the anonymizer. On the other hand, the generalization shows that these solutions are unlikely to exist: whatever number of features the anonymiser is going to protect, it might always exist an adversary leveraging the knowledge of a further feature. Hence, our results open a new perspective on the definition of the privacy goals and the strategy to be adopted to reach those goals for LBSs. They also encourage continuing the effort of comprehension that have already produced some important clarifications and distinctions regarding the usage of the k-anonymity approach in preserving privacy for LBSs [33].

This work provides several contributions to privacy in Location Based Services. We first model footprints knowledge as an hypercube—the features being its dimensions. Then, we show how user privacy can be violated leveraging structured knowledge on dimensions of the cube, with respect to the solutions in [39, 40]. In particular, we investigate on the provided privacy considering a different, more realistic adversary model. Note that it can also be weaker (with respect to [39, 40]) in terms of the amount of users information available, but still effective. We introduce our claim through a practical example; we then support and verify the claim analyzing a real data set of GPS traces.

The rest of the chapter is organized as follows. Section 2 describes the related work in the area. Section 3 defines the notion of time and presents the threat model and the feeling-based privacy model. Section 4 shows how user privacy can be violated applying our considerations; we support our claim with both analysis and a practical example. Section 5 discusses and compares results from the analysis of a real data set. Finally, Sect. 6 reports some concluding remarks.

2 Related Work

Privacy concerns is the main issue that hinder a wide adoption of LBSs [21, 28, 35]. Providing privacy for LBSs is not an easy task. In fact, the specific characteristic of mobile devices (for example, their mobility or the fact that they connect always to the same network operator) make privacy solutions already designed for other environments—like the ones for databases based on k-anonymity [22, 34, 36]—not directly suitable also for this context.

User mobility is the main issue for anonymization of LBSs. In fact, mobile users ask for LBSs from different locations that correspond either to their current position or to other positions of their interest. The first approach [14] for location anonymity aimed at applying the k-anonymity concept. The proposal was to reduce the accuracy of the definition of the user location (defined by both space and time) when asking for a LBS. The aim of reducing this accuracy was to cloak the requesting user within $k - 1$ other users, present in a broader area and considering a broader time frame. However, increasing the area would lead to a coarser service, while increasing the time frame would lead to a delay of the user's request.

Several works leveraged on the basic concept introduced in [14]. For example, the CliqueCloak algorithm [12] aims at minimizing the size of the cloaking area, while allowing the user to specify the value of k. However, this solution is practicable only for small values of k and requires a high computation overhead. The work in [38] generates a cloaking area in polynomial time and also considers attacks that correlate periodic location updates. The possibility of selecting a specific k is also leveraged in [27], without considering the minimization of the cloaking area. Further work [5] provided also solution for mobile peer-to-peer environment, where the cloaking area is determined in a distributed way.

All the above work does not explicitly consider the fact that nodes move and their location-related request might be correlated. This issue has been first addressed by some works [3, 17] with the aim of avoiding nodes tracing. However, these solutions were not developed having LBSs privacy in mind. In fact, they all report the actual user location. In particular, the work in [3] introduced the concept of mix zone—a zone where nodes avoid reporting their locations and exchange their identification instead. The aim of a mix zone is to make it hard for an adversary to correlate the pseudonym that a node used before entering the mix zone, and its pseudonym once it is out of the mix zone. Selfish behaviour of the nodes in mix zones has also been considered recently [9], as well as how pseudonyms aging affects privacy [10]. An idea similar to the one of mix zone is *path confusion* [17, 18]—pseudonyms are exchanged between nodes that have paths close to each other. The mix zone concept is also applied in [11] to protect the location privacy of drivers in a VANET scenario. The idea is to combine mix zones with mix networks that leverage on the mobility of vehicles and the dynamics of road intersections to mix identifiers. The solution proposed in [19] requires that each LBSs request comes together with at most $k - 1$ dummy requests that simulate the movement of nodes. However, the dummy traces do not take into consideration the actual geography of the area where the corresponding

dummy user is expected to be—such an anomaly can hence let the adversary identify the dummy requests. Trajectory anonymization is also considered in [4], increasing the cloaking area to include exactly $k - 1$ other users. Unfortunately, continuously increasing the cloaking area degrades the precision of the LBSs.

A slightly different problem, that is avoiding reporting information about sensitive areas (e.g., a night club, a political gathering), has also been addressed [15]. Here anonymization is achieved using areas instead of users. In fact, the reported location should include k sensitive areas instead of k users. Similarly, the framework proposed in [6] provides obfuscation of sensitive semantic locations based on the privacy preference specified by each user. The solution uses a probabilistic model of space—the semantic locations being expressed in terms of spatial features—and does not take time into account. The solution proposed in [13] aims to avoid reporting the user location. The technique applies a Private Information Retrieval protocol to let the user of the service to download directly the LBSs information without requiring a trusted anonymiser. However, as the amount of data to be downloaded by the user depends on the total amount of data stored by the service provider, it may be impractical for a mobile device. Other solutions based on obfuscations were presented in [1, 2]. A problem strictly related to the protection of the user location privacy is the quantification of the "privacy level" guaranteed by several solutions. The solution in [18] quantifies location privacy as the duration over which an attacker could track a subject. The expected error in distance between a person's current location and an attacker's uncertain estimate of that location is used in [17]. The number of users k represents the level of privacy in [14] where k-anonymity is introduced for location privacy.

Recently, the analysis proposed in [33] highlights some flaws in the usage of the k-anonimity approach for LBS. In [33], the anonymization of a user (that requires a service based on her location) includes two distinct aspects: query anonymity and location anonymity. Achieving query anonymity implies the impossibility to link a query to the user identity; achieving location anonymity implies the impossibility to link the location, at a given time, to a user identity. These are two different properties. In fact, cloaking can help decoupling a query and a user (query anonymity). This is not the case for hiding the location (location privacy), as it does not seem to depend on k.

Other works derive metrics from information theory [31]. For instance, entropy is the privacy quantifier used in [3, 40]. Whatever location privacy metric is adopted, it is maximized if no one knows a subject's location. Hence, the majority of the proposed solutions can be considered a trade-off between location privacy and quality of service.

In this work we show that leveraging structured knowledge (even partial) on footprints, provides an adversary with a powerful tool to compromise privacy in LBSs. Preliminary investigations, considering the time dimension, appeared in [24, 25]. In particular, in [24, 25] we show an application of this concept by compromising the privacy claimed in [39, 40], where the feeling based model is introduced. Being it a reference also for this chapter, we recall the feeling based model in Sect. 3.2. We also observe that our preliminary findings in [24, 25] are consistent with the

recent proposal in [32], where time is considered one of the aspects to take into account to protect user's location. To show how the considerations introduced in [24, 25] for the time variable can be generalized, we use one particular characteristic of a mobile user: her mode of transportation. This characteristic has been investigated in some works on post-processing of GPS raw data [30, 43]. In particular, as a result of the post-processing techniques described in [41–43], the authors made available a data set of users GPS traces, labeled with their inferred transport modes. We used this data set for the experiments reported in this chapter.

3 Preliminaries and Notation

In this section, we introduce models and definitions used in the chapter. Section 3.1 introduces the system model. Section 3.2 gives an overview of the solutions proposed in [39, 40], while the threat model description can be found in Sect. 3.3. The formalization of the notion of time, applied to time-related concepts used in this work, concludes the section.

3.1 System Model

We consider the same system architecture used in [39, 40]. We assume mobile nodes (users) communicating with location-based services (LBSs) providers through a central anonymity server—the location depersonalization server (LDS)—which is considered trusted. The LDS server is managed by some mobile service provider allowing the (mobile) users to access to wireless communications. The provider offers the depersonalization service as an added value service and supplies the LDS server with an initial footprints database derived from users phone calls.

3.2 Feeling Based Privacy Model

The aim of the work in [40] is to provide location privacy protection for users requesting a location-based services with enhanced features with respect to the standard k-anonymity model. The cited privacy model introduces the concept of feeling-based privacy, based on the intuition of privacy being mainly a matter of feeling. The user is allowed to express a privacy requirement by specifying a spatial region in which she would feel comfortably cloaked (public region). Their solution then transforms the intuitive notion of user privacy feeling, in a quantitative evaluation of the level of protection provided, using the user specified region. They define the entropy of a spatial region to measure the popularity of that region. This popularity is then used as the quantity describing the user privacy requirement: the popularity of the

location disclosed by the anonymiser on behalf of the user, is required to be at least that of the specified public region. Formally, they provide the following definitions.

Definition 1 Entropy. Let R be a spatial region and $S(R) = \{u_1, u_2, \ldots, u_m\}$ be the set of users having footprints in R. Let n_i $(1 \leq i \leq m)$ be the number of footprints that user u_i has in R, and $N = \sum_{i=1}^{m} n_i$. The entropy of R is defined as $E(R) = -\sum_{i=1}^{m} \frac{n_i}{N} \cdot \log \frac{n_i}{N}$.

Definition 2 Popularity. The popularity of R is defined as $P(R) = 2^{E(R)}$.

The entropy is used to address the problem of the possible dominant presence of some users in a certain region. This phenomenon makes the number of visitors of a region not sufficient to quantify its popularity. The property that $P(R)$ is higher if m is larger is preserved even using entropy: a region is more popular if it has more visitors. Also, a skewed distribution of footprints significantly reduces the $P(R)$ with respect to a symmetric distribution. The entropy is also intended by the authors as the amount of additional information needed for the adversary to identify the service user from $S(R)$ when R is reported as her location in requesting an LBSs.

3.3 Threat Model

In this section we present our threat model. We assume the adversary being present from time t_0, that is from the system deployment. Hence, we observe that the adversary may coincide with the LBSs provider. In fact, it could be highly interested in exploiting the location knowledge (historical, such as in [20]) of the LDS anonymiser—potentially motivated by commercial or marketing purposes. Thus *ADV* and LBSs will be used interchangeably throughout the chapter.

Some existing techniques use current location of k neighbours of the service requester, to protect from the adversary and to calculate the cloaking area. These techniques protect the anonymity of the service users but not their location privacy. An adversary identifying the users in the cloaking area knows their locations as it is aware of their presence in the cloaking area at the time of the service request.

The idea to use footprints, that is historical data, makes the adversary weaker as it is not able to know neither who requested the service nor who was really there at the time of the service request. This core idea, introduced in [39] and applied by the same authors to mobile user's trajectory in [40], also conveys another implicit assumption: the indistinguishability for the *ADV* between current and historical visitors of the cloaking area. This is equivalent to assume that *ADV* can not have instantaneous access to current users location data. If this would be the case, the usage of historical locations would not be suitable to compute the cloaking box for depersonalization.

As an example, let us suppose the LDS reporting a cloaking area for a user, based on a five (historical) footprints. If the user is the only one actually in that area and the LBSs know the user location at each time instant, our adversary would immediately identify the service requester. Thus, we also assume the users location

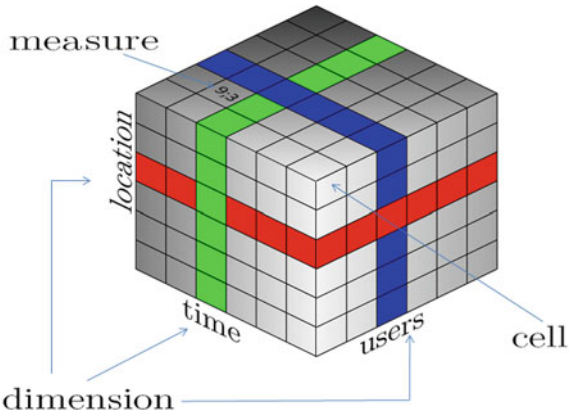
knowledge held by the adversary to be the footprints information provided by the LDS anonymiser.

In this work we conceptually model the footprints data as an hypercube (data cube for short), a model introduced by the OLAP (on-line analytical processing) community. The data cube is a data abstraction to view data of an existing database, at various level of detail, and focusing on various combinations of its attributes. The cube is created considering a subset of attributes of the data stored in a database. The attributes describing each data item are selected as functional attributes or *dimensions* of the cube; the attributes whose values are of interest for analysis, are chosen to be measure attributes; some dimensions can be hierarchical: for instance multiple hierarchies exists for time dimension (year-quarter-month, or week-day). As a real example, the user footprints (the measure attribute) could be structured with respect to the dimensions of time, age, and gender of the users and their mode of transportation (functional attributes). Figure 1 shows an example of data cube. A tuple of dimension-value defines a cell of the data cube. The cell pointed out in Fig. 1 denotes a certain user, in a certain location at a certain time. Each cell contains values of measure attributes. As an example, in Fig. 1, the case is depicted where a pair of measure attributes (9;3) is associated to a cell. Measure attributes can be further aggregated applying functions. The most common functions used for aggregation are sum, min, max and average, but any function can actually be applied. For example, using the number of footprints as the base measure attribute, it is possible to calculate any footprints-based privacy metrics, like the entropy in [40], as aggregated measure. Different views on the footprints data can be obtained from the cube applying data cube operations: slice, dice, drill-down and roll-up being the most important ones [16]. Slicing is analogous to the selection operation in relational algebra and is the operation of selecting the dimensions used to define a view on the cube. The highlighted stripes in Fig. 1 are examples of slicing operations. Dicing is analogous to the projection operation in relational algebra and is the operation of selecting actual values on a dimension. Roll-up and drill-down are the operations that allow to perform analysis across a hierarchical dimension, increasing and decreasing granularity, respectively.

The intuition underneath this kind of modeling is that considering different features, induces different partitions on the footprints data set. As a consequence, the privacy metric (i.e. entropy) calculated on some partitions may be different from the metric calculated on other partitions or on the whole set (See Theorem 1). In this work we study the interactions and relations between the anonymiser and the adversary footprints data cubes. We also investigate which properties must hold in order to guarantee privacy.

Our model formalization leverages the multidimensional data cube model introduced by Datta and Thomas in [37]. They defined an abstract structure, the "cube", and an algebra to operate on it. We choose their model, among the variety available, as it separates structure (cube schema) and content (cube instance). This allows to formalize how the structure of knowledge (i.e. user footprints) impacts the privacy guarantees of models based on that knowledge. Here we report only the portion of the formal model that suffices to the purposes of our work.

Fig. 1 Example of data cube



Definition 3 Cube Schema. A cube is a logical structure defined by a 5-tuple $\langle D, M, A, f, O\rangle$ where:

- D is a set of d dimensions $D = \{d_1, d_2, \ldots, d_d\}$ where $d_i \in D_i$, $1 \leq i \leq d$, that is d_i is a dimension name extracted from a domain D_i;
- M is a set of r measures $M = \{m_1, m_2, \ldots, m_r\}$ where $m_i \in M_i$, $1 \leq i \leq r$, that is m_i is a measure name extracted from a domain M_i;
- $D \cap M = \varnothing$, i.e. the set of dimensions D and the set of measures M are disjoint;
- A is a set of t attributes $A = \{a_1, a_2, \ldots, a_t\}$ where $a_i \in A_i$, $1 \leq i \leq t$, that is a_i is an attribute name extracted from a domain A_i;
- O is a set of partial orders on the set A;
- f is a one-to-one mapping $f : D \longrightarrow 2^A$, that associates a set of attributes to each dimension. The mapping is such that attribute sets corresponding to dimensions are pairwise disjoint, i.e., $\forall i, j, i \neq j, f(d_i) \cap f\left(d_j\right) = \varnothing$

A cube instance is obtained from a cube schema assigning values to the measures along all dimensions.

Definition 4 Cube Instance. A cube instance is defined by a 7-tuple $\langle D, M, A, f, O, V, g\rangle$ where the elements D, M, A, f, O are inherited from the cube schema definition whereas every element $v \in V$ is a r-tuple $\langle \mu_1, \mu_2, \ldots, \mu_r\rangle$ and each μ_i is an instantiation of the ith measure m_i; g is a mapping $g : D_1 \times D_2 \times \cdots \times D_d \longrightarrow V$.

Intuitively the g mapping indicates which values are associated with specific points in the multidimensional space (cells for short). In the literature, the cube cells are also denoted as a set of pairs of the form $\langle$address, content$\rangle$ where the address is specified by the cube dimensions and the content by the values of measure attributes.

Definition 5 Predicate P. A predicate P is a well-formed formula in first-order predicate logic.

- an atomic predicate, denoted by p, is a restriction on the domain of a single attribute (i.e. day = Monday).

- a compound predicate is a logical expression of atomic predicates. It assumes the form: $P = p_1 \langle \text{ op } \rangle p_2 \langle \text{ op } \rangle \ldots \langle \text{ op } \rangle p_k$ where $\langle \text{ op } \rangle$ represents logical operators ($\wedge$ (and), $\vee$ (or), $\neg$ (not), $\Longrightarrow$ (implies) and $\Longleftrightarrow$ (equivalent to)).

On this logical Cube structure, Datta and Thomas propose several operators. Among them, we only report the restriction operator that reduces the values on one or more dimensions. The interested reader can refer to [37] for the complete operators definitions. The algebra of the restriction operator is defined as follows:

Definition 6 Restriction σ.

- Input: a cube $C_I = \langle D, M, A, f, O, V, g \rangle$ and a predicate P;
- Output: a cube $C_O = \langle D_O, M_O, A_O, f_O, O_O, V_O, g_O \rangle$ where $D_O = D$, $M_O = M$, $A_O = A$, $f_O = f$, $V_O \subseteq V$ and $g_O = g_P$, g_P being a mapping such that every element of $g_P^{-1}(V_P)$ satisfies P;
- Notation: $\sigma_P(C_I) = C_O$.

Notice that the slicing and dicing operations, whose scope is to reduce the cube dimensionality, can be defined through the restriction operator, σ. With an atomic predicate, the restriction operator implements the slicing operation while with a compound predicate it realizes the dicing one. Figure 1 illustrates the footprints knowledge abstracted by a cube and, in the highlighted portions, the results of slice and dice operations with respect to one or more dimensions of the cube.

With this formal model in mind, the anonymiser knowledge (LDS knowledge) is a data cube denoted with K whereas the adversarial knowledge of the adversary, in our threat model, is a data cube denoted with $\hat{K}$. The adversary with knowledge $\hat{K}$ is denoted with $ADV^{\hat{K}}$. Also, the threat model introduced in our previous work [25], can be rethought as an instance of this generalization where: the anonymiser knowledge is a cube considering only the spatial dimension, while the adversarial knowledge is a cube with a single additional hierarchical feature, the time. We will study the relation between the two cubes, K and $\hat{K}$, and the impact on privacy guarantees in Sect. 4.1. We can observe that the knowledge of $ADV^{\hat{K}}$ might be lower than the anonymiser knowledge. In fact, $ADV^{\hat{K}}$ could know footprints information regarding just a portion of the cube dimensions.

Table 1 summarizes the notation used in this work. The notion of time used in the discussions and examples is defined as follows.

Formalizing Time. Consistently with the literature [32], we consider a discrete time-line, starting from time t_0—this time corresponding to the deployment of the system. The notion of time is referred as a hierarchical dimension composed of: the smallest measurable unit in our discrete line (time unit), a predetermined number of contiguous units (time period), a time interval of predetermined length (time slice), and the set of the i-th time slice of each period (time frame). For completeness, we report Fig. 2 specifying that the notation in the figure will be only referred in the numerical example of Sect. 4.2.

For a practical discussion, time parameters to be fixed are thus the length of the period and that of the time slice. As exemplified in Fig. 2a, time period is composed

Table 1 Notation table

R	A spatial region
K	Footprints knowledge cube
$\hat{K}$	Adversarial footprints knowledge cube
P	Logical predicate
$S(R)$	Set of users having footprints in R
$E(R)$	Entropy of region R
$P(R)$	Popularity of region R
d_j	Dimension (feature) $d_j \in \{d_1, d_2, \ldots, d_d\}$
$E(R, \sigma_P(K))$	Entropy of region R, w.r.t. restriction σ on cube K by predicate P
$P(R, \sigma_P(K))$	Popularity of region R, w.r.t restriction σ on cube K by predicate P
u_i	Generic i-th user of a set of users, $1 \leq i \leq m$, $m \in \mathbb{N}$
$u_{i,\sigma_P(K)}$	Generic i-th user who have footprints in R w.r.t restriction σ on cube K by predicate P
n_i	Number of footprints of user u_i in R
$n_{i,\sigma_P(K)}$	Number of footprints of user u_i in R w.r.t. restriction σ on cube K by predicate P
N	Total number of footprints in a region R

of time slices that are time intervals of predetermined length (ℓ and s respectively). We denote time slice j of time period p with T_j^p. A time frame is defined as the set obtained as the union of the j-th time slice of each period, indicated in Fig. 2a

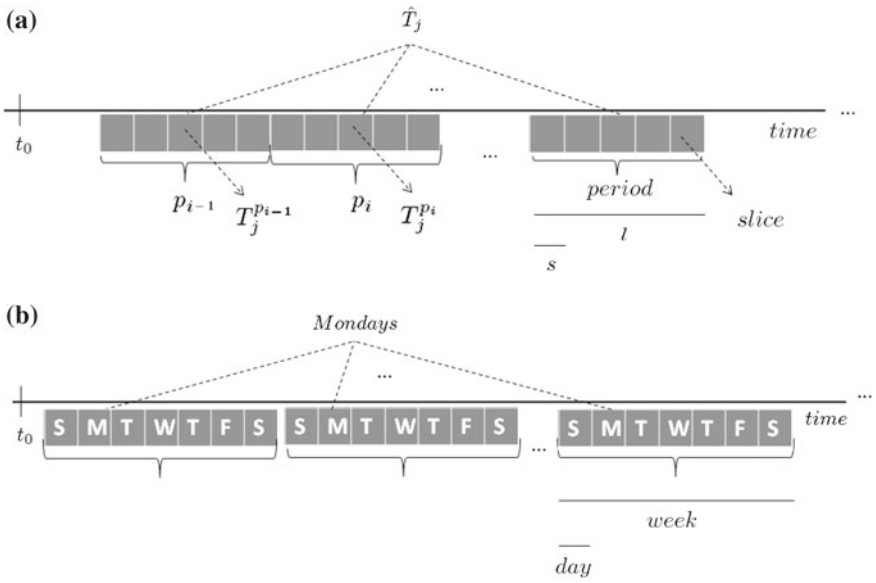


Fig. 2 Time formalization. **a** Time definitions. **b** Time example

with $\hat{T}_j$. As an example, consider Fig. 2b. If we fix the period length to be one week, and the slice length to be one day, the period p is set to be the p-th week, $T_1^p = Sunday, T_2^p = Monday, \ldots, T_7^p = Saturday$ represent the days of the p-th week.

4 Facing a Multi-dimensional Adversary

In this section, we aim to investigate on the privacy guaranteed by the existing footprints-based solutions [39, 40] when facing a multi-dimensional adversary $ADV^{\hat{K}}$. Section 4.1 introduces the adversary model used and an example showing how user privacy can be violated. Section 4.2 provides an evaluation of the adversary effectiveness against the privacy guarantees of the protocol in [40].

4.1 The Feature-Aware Adversary Model

Our adversary model is motivated by the fact that user location privacy may be highly influenced by the context features. We have already analyzed in our previous work [25] the influence of the time frames on user's location privacy. In this work we show how the process can be extended to any general dimension we take into account. Considering the time we might refer to several real scenarios: a theatre is a physical place where users concentrate only on particular days and in specific time frames; restaurants are most likely to be crowded at lunch and dinner time; and, office buildings are supposed to be almost empty during night. Considering the mobility of the user we can refer to different modes of transportation: walking, biking, driving or using public transportation. We could enunciate many other scenarios further varying the user profile dimension (e.g.: age, gender, profession).

All these scenarios originate from different views over reality and are obtained focusing on different aspects. Thus, modeling these scenarios results in producing an associated cube-schema—each aspect being a dimension on its own. Especially, we aim at formally capturing the intuitive notion of a different structured knowledge of an adversary with respect to the knowledge of the anonymiser. We thus assume that the anonymiser and the adversarial knowledge are related to each other as follows. The schemes underneath the cube instances $\hat{K}$ and K satisfy the conditions: (i) $D \subset \hat{D}$; and, (ii) $\sigma_{\hat{P}}(\hat{K}) \subseteq K$, $\forall \hat{P}$, $\forall \sigma$. The first condition captures the concept of more structured knowledge—the adversarial cube having at least one more dimension with respect to the anonymiser. The second condition captures the concept that the new adversary considered, $ADV^{\hat{K}}$, can be even weaker than ADV—her knowledge being at most equal to the anonymiser one.

We show that with the knowledge held by $ADV^{\hat{K}}$, the LDS is no more able to guarantee to users the claimed level of privacy. Further, we also show scenarios where

the entropy of the user public region is actually less than the entropy calculated by the LDS. Therefore, the adversary may need less effort—with respect to what assumed by the LDS—to identify the user. We will show that $ADV^{\hat{K}}$ may be effective even if provided with less knowledge. This, as we formally show at the end of this section, is due to the fact that context features (adversarial additional dimensions) severely affect the entropy and the popularity of a cloaking region. This may result in a reduced amount of additional information needed by the adversary to identify the service user (see Sect. 3.2).

Let us denote the cube representing the anonymiser footprints data with K. The anonymiser cube has:

- $D = \{user, location\}$;
- $A = \{id\_user, latitude, longitude, height\}$;
- $M = \{number\_of\_footprints\}$;
- $f(user) = \{id\_user, requested\_privacy\}$;
- $f(location) = \{latitude, longitude, height\}$;

Let us consider two different cubes representing the adversarial footprints knowledge. The first cube, $\hat{K}_1$, considers an adversary taking the time dimension into account, like the one in [25].

- $\hat{D}_1 = \{user, location, time\}$;
- $\hat{A}_1 = \{id\_user, requested\_privacy, latitude, longitude, height, t\_slice, t\_period, t\_frame\}$;
- $\hat{M}_1 = \{number\_of\_footprints\}$;
- $\hat{f}_1(user) = \{id\_user, requested\_privacy\}$;
- $\hat{f}_1(location) = \{latitude, longitude, height\}$;
- $\hat{f}_1(time) = \{t\_slice, t\_period, t\_frame\}$;
- $\hat{O}_{time} = \{\langle t\_slice, t\_period\rangle, \langle t\_slice, t\_period, t\_frame\rangle\}$;

The second cube, $\hat{K}_2$, models an adversary taking into account both time and the user professional role

- $\hat{D}_2 = \{user, location, time, professional\_role\}$;
- $\hat{A}_2 = \{id\_user, requested\_privacy, medical\_specialty, latitude, longitude, height, t\_slice, t\_period, t\_frame\}$;
- $\hat{M}_2 = \{number\_of\_footprints\}$;
- $\hat{f}_2(user) = \{id\_user, requested\_privacy\}$;
- $\hat{f}_2(location) = \{latitude, longitude, height\}$;
- $\hat{f}_2(time) = \{t\_slice, t\_period, t\_frame\}$;
- $\hat{f}_2(professional\_role) = \{medical\_specialty\}$;
- $\hat{O}_{time} = \{\langle t\_slice, t\_period\rangle, \langle t\_slice, t\_period, t\_frame\rangle\}$;

Definition 7 Entropy with respect to $\sigma_{\mathbf{P}}(\mathbf{K})$. Let R be a spatial region and $S(R, \sigma_P(K))$ be the set of users who have footprints in R, if observed with respect to the restriction σ_P of the cube instance K. That is: $S(R, \sigma_P(K)) = \{u_{1,\sigma_P(K)}, u_{2,\sigma_P(K)}, \ldots, u_{m,\sigma_P(K)}\}$, where $n_{i,\sigma_P(K)} (1 \leq i \leq m)$ is the number of

footprints that user u_i has in R with respect to $\sigma_P(K)$ and $N_{\sigma_P(K)} = \sum_{i=1}^{m} n_{i,\sigma_P(K)}$. We define the entropy of R with respect to $\sigma_P(K)$ as:

$$E(R, \sigma_P(K)) = -\sum_{i=1}^{m} \frac{n_{i,\sigma_P(\mathbf{K})}}{N_{\sigma_P(\mathbf{K})}} \cdot \log \frac{n_{i,\sigma_P(K)}}{N_{\sigma_P(K)}}.$$

Definition 8 Popularity with respect to $\sigma_{\mathbf{P}}(\mathbf{K})$. We define the popularity of R with respect to $\sigma_P(K)$ as $P(R, \sigma_P(K)) = 2^{E(R,\sigma_P(K))}$.

We use the following example to support our discussions and to compare with the privacy model in [39, 40].

Example. Let us consider a user, Alice, requesting a LBS from her office building. She feels her privacy is preserved when specifying her office as the public region. Alice's office is an health center and the different specialist doctors are organized on work shifts. Part of the doctors are on a morning shift and the remaining ones on an afternoon shift with the only exception of the dentist that is available all day long. The medical staff is composed of 3 dentists, 2 oculists and 1 psychologist. In our formal setting, the domain of attribute *medical_specialty* is the set of values {dentist, oculist, psychologist}. As the doctors share two rooms, no more than one doctor per specialties is at the office. Thus, let us suppose that in the morning there are one oculist and one dentist while in the afternoon one psychologist and one dentist. We notice that this type of external information regarding the organization of the health center can be gained very easily just looking at the door plate. Let us consider $m = 6$ users ($u_1, u_2, u_3, u_4, u_5, u_6$) for the region corresponding to Alice's office (later on also referred to as region R_1), each of them having 16 footprints in the LDS footprints database—the cube K. This scenario is depicted in Fig. 3. The corresponding footprints data for $u_1, u_2, u_3, u_4, u_5, u_6$ are provided and highlighted in the first column of Table 2a, b, c, d, e and f, respectively.

Data in Table 2a represents the footprints information used by the LDS to calculate the entropy and the popularity of Alice's office. The results of the calculation determine a corresponding spatial region R_j (column labels in Table 2) used to cloak the user location. Hence, Table 2a also represents the knowledge of ADV. Table 2b and 2c instead represent the structured knowledge $\hat{K}_1$ of an adversary $ADV^{\hat{K}_1}$, that is the same information of ADV when taking time into account. In particular, in Table 2b we consider a restriction using the predicate $\hat{P}_0$ that selects all the footprints data in $\hat{K}_1$ satisfying the condition $t\_frame = morning$; in Table 2c we use the predicate $\hat{P}_1$ that selects all the footprints data in $\hat{K}_1$ satisfying the condition $t\_frame = afternoon$.

Table 2d, e and f represents the structured knowledge $\hat{K}_2$ of an adversary $ADV^{\hat{K}_2}$, that is the same information of ADV when taking into account two features: time, with the same granularity of $\hat{K}_1$ (morning and afternoon time frames), and the professional role. Here the logical predicates used to restrict the $\hat{K}_2$ knowledge are: $\hat{P}_2$ in Table 2d, that considers only one of two features (the professional role), restricting footprints with the condition $medical\_specialty = dentist$; $\hat{P}_3$ in Table 2e and $\hat{P}_4$ in Table 2f restricting on the two dimensions with the condition $t\_frame = morning \wedge t\_frame =$

Table 2 ADV, $ADV^{\hat{K}_1}$ and $ADV^{\hat{K}_2}$ table data

User	R_1	R_2	
(a) ADV: daily			
u_1	**16**	9	
u_2	**16**	14	
u_3	**16**	20	
u_4	**16**	25	
u_5	**16**	10	
u_6	**16**	18	
$E(R)$	2.58	2.49	
$P(R)$	6	5.64	
User	R_1	R_2	R_3
(b) $ADV^{\hat{K}_1}$*: morning*			
$u_{1,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**16**	4	8
$u_{2,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**16**	8	8
$u_{3,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**0**	0	8
$u_{4,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**16**	25	8
$u_{5,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**16**	10	8
$u_{6,\sigma_{\hat{P}_0}(\hat{K}_1)}$	**0**	0	0
$E(R, \sigma_{\hat{P}_0})$	2	1.7	2
$P(R, \sigma_{\hat{P}_0})$	4	3.24	4
User	R_1	R_2	R_3
(c) $ADV^{\hat{K}_1}$*: afternoon*			
$u_{1,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**0**	5	8
$u_{2,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**16**	6	8
$u_{3,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**16**	20	8
$u_{4,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**0**	0	8
$u_{5,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**0**	0	8
$u_{6,\sigma_{\hat{P}_1}(\hat{K}_1)}$	**16**	18	8
$E(R, \sigma_{\hat{P}_1})$	1.58	1.77	2
$P(R, \sigma_{\hat{P}_1})$	3	3.4	4
User	R_1	R_2	R_3
(d) $ADV^{\hat{K}_2}$*: dentist*			
$u_{1,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**16**	12	8
$u_{2,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**16**	22	8
$u_{3,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**16**	14	8
$u_{4,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**0**	0	0

(continued)

Table 2 (continued)

User	R_1	R_2	
$u_{5,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**0**	0	0
$u_{6,\sigma_{\hat{P}_2}(\hat{K}_2)}$	**0**	0	0
$E(R, \sigma_{\hat{P}_2})$	1.58	1.53	1.58
$P(R, \sigma_{\hat{P}_2})$	3	2.9	3
User	R_1	R_2	R_3
(e) $ADV^{\hat{K}_2}$: *morning*, *dentist*			
$u_{1,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**8**	4	8
$u_{2,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**16**	14	8
$u_{3,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**0**	9	8
$u_{4,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**0**	0	0
$u_{5,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**0**	0	0
$u_{6,\sigma_{\hat{P}_3}(\hat{K}_2)}$	**0**	0	0
$E(R, \sigma_{\hat{P}_3})$	0.92	1.43	1.58
$P(R, \sigma_{\hat{P}_3})$	1.89	2.69	4
User	R_1	R_2	R_3
(f) $ADV^{\hat{K}_2}$: *afternoon*, *dentist*			
$u_{1,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**8**	8	8
$u_{2,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**0**	8	8
$u_{3,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**16**	5	8
$u_{4,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**0**	0	0
$u_{5,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**0**	0	0
$u_{6,\sigma_{\hat{P}_4}(\hat{K}_2)}$	**0**	0	0
$E(R, \sigma_{\hat{P}_4})$	0.92	1.96	2
$P(R, \sigma_{\hat{P}_4})$	1.89	3.88	4

dentist and $t\_frame = afternoon \wedge medical\_specialty = dentist$, respectively. Each table is provided with additional column data to show that both the entropy and the popularity depend on footprints distribution among visitors. In fact, it is possible to check that in each reported scenario the total number of footprints per user remains unchanged. Let us take the values of entropy and popularity in Table 2a as reference point to evaluate both the entropy and the popularity: (i) for each data column in Table 2b and c—considering the adversarial knowledge $\hat{K}_1$; (ii) for each data column in Table 2d, e and f—considering the adversarial knowledge $\hat{K}_2$. As it is shown in Table 2a column 1, the maximum is obtained from a uniform distribution of footprints (column 1). We can observe that a more structured knowledge, like that of $ADV^{\hat{K}_1}$ in

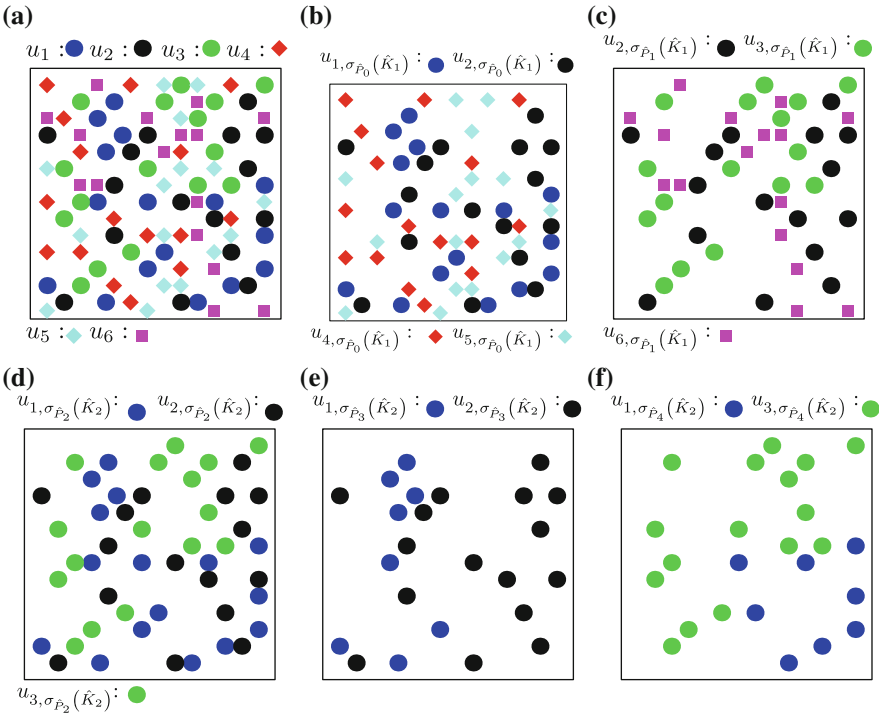


Fig. 3 $K, \hat{K}_1$ and $\hat{K}_2$ knowledge. **a** ADV: daily. **b** $ADV^{\hat{K}_1}$: morning. **c** $ADV^{\hat{K}_1}$: afternoon. **d** $ADV^{\hat{K}_2}$: dentist. **e** $ADV^{\hat{K}_2}$: morning, dentist. **f** $ADV^{\hat{K}_2}$: afternoon, dentist

Table 2b and c may result in the following possible scenarios: (i) $ADV^{\hat{K}_1}$ entropy and popularity values are strictly less than that of ADV. This is the case for the first and the second data columns in Table 2c and for the first column in Table 2b, compared to the corresponding columns in Table 2a; (ii) $ADV^{\hat{K}_1}$ entropy and popularity values are equal to that of ADV (see Table 2b and c column 3); (iii) $ADV^{\hat{K}_1}$ entropy and popularity values are greater than that of ADV. This is the case for the second column in Table 2b with entropy 1.51—greater than the corresponding 1.49 in Table 2a.

In the following, we formally prove that an anonymiser using the aggregated data can guarantee the level of privacy requested by the user only if it is facing the adversary ADV. In fact, we prove that when the anonymiser is facing $ADV^{\hat{K}}$, the following two cases can also happen: (i) the anonymiser is not able to guarantee the user to be protected with the requested level of privacy; (ii) the anonymiser is degrading the accuracy of the location information for the LBSs. We observe that in our formal setting, the restriction of the anonymiser knowledge K with respect to the predicate P that selects users locations in the public region coincides with the knowledge of the adversary ADV considered in the feeling based model. Thus, $E(R, \sigma_P(K)) = E(R)$ and the two notations are interchangeably used.

Theorem 1 *Given a spatial region R and footprints data $\sigma_{\hat{P}}(\hat{K})$ related to the restriction of cube $\hat{K}$ w.r.t. predicate $\hat{P}$, there might exist footprints distributions such that $E(R, \sigma_{\hat{P}}(\hat{K})) \neq E(R)$.*

Proof The proof is a direct consequence of the two following cases. *Case 1* If $n_{i,\sigma_{\hat{P}}(\hat{K})}$ satisfies $n_{i,\sigma_{\hat{P}}(\hat{K})} \leq n_i \cdot \frac{N_{\sigma_{\hat{P}}(K)}}{N}$, then $E(R, \sigma_{\hat{P}}(\hat{K})) \leq E(R)$. In fact, the condition can be rewritten as: $\frac{n_{i,\sigma_{\hat{P}}(\hat{K})}}{N_{\sigma_{\hat{P}}(\hat{K})}} \leq \frac{n_i}{N}$. Since the log function is monotonically increasing, $\log \frac{n_{i,\sigma_{\hat{P}}(\hat{K})}}{N_{\sigma_{\hat{P}}(\hat{K})}} \leq \log \frac{n_i}{N}$. As a consequence, $E(R, \sigma_{\hat{P}}(\hat{K})) \leq E(R)$.

Case 2 If $n_{i,\sigma_{\hat{P}}(K)}$ satisfies $n_{i,\sigma_{\hat{P}}(\hat{K})} > n_i \cdot \frac{N_{\sigma_{\hat{P}}(\hat{K})}}{N}$, then $E(R, \sigma_{\hat{P}}(\hat{K})) > E(R)$. The proof is similar to the proof of Case 1.

Case 1 shows that with a feature-aware adversary, $ADV^{\hat{K}}$, the LDS is not always able to guarantee the level of privacy requested by the user. This happens when $E(R, \sigma_{\hat{P}}(\hat{K})) < E(R)$. In fact, if this is the case, the region R does not achieve an entropy at least equivalent to the public region specified by the user in order to meet her privacy requirement. Case 2 shows that with a feature-aware adversary, $ADV^{\hat{K}}$, the LDS is not always able to guarantee the maximum level of accuracy for the LBSs service requested by the user. This happens when $E(R, \sigma_{\hat{P}}(\hat{K})) > E(R)$. If this is the case, the LDS introduces a loss in service accuracy—since a region larger than the necessary is used to guarantee the user requested level of privacy.

4.2 Evaluating the Adversary Effectiveness

In this section, we evaluate the adversary effectiveness against the privacy guarantees of the protocol in [40]—showing the influence of the features determining the views on footprints data. To do so, we first formalize the adversary effectiveness in terms of probability and then we plot the analytical results of some example data. The aim of the graph is to show how footprints distribution affect the entropy values used to measure the required adversary effort. We remind that the entropy is a measure for the adversary effort needed to compromise the user privacy. Let us assume the user selected a desired level of privacy (entropy). On the one hand, if the anonymiser behaves in such a way that the effort required to $ADV^{\hat{K}}$ to compromise privacy is less than the expected one, the anonymiser is failing in guaranteeing the claimed level of privacy. On the other hand, each time the actual level of entropy for $ADV^{\hat{K}}$ is greater than the one sufficient for guaranteeing the user's chosen level of privacy, the anonymiser is decreasing the quality of the LBSs. This concept can be formally expressed through the cumulative distribution function of the entropy considered as a random variable. In fact, varying the logical predicates and applying the restriction operator to obtain the adversary knowledge, we obtain different users footprints

distributions and as consequence an entropy value for each of them. This can also be rewritten as the entropy being a random variable $\mathbb{Y}$ that is a function F of a vector of random variables $\mathbb{X} = (X_1, \ldots, X_m)$ representing the footprints per user (given a region R).

Assuming the footprints probability distribution known, the following equation captures the likelihood of the cases in which the anonymiser is decreasing the quality of LBSs services:

$$Pr[E(R, \sigma_{\hat{P}}(\hat{K})) > E(R)] = Pr[\mathbb{Y} > y] = \int_{\mathbf{x}'}^{\mathbf{x}''} F(\mathbf{s})\, \mathrm{d}\mathbf{s} \tag{1}$$

with $\mathbf{x}'$ s.t. $F(\mathbf{x}') = y$ and $F(\mathbf{x}'') = \log_2 m$. The entropy variation is in the logical predicates used and $\mathbf{y}$ represents the actual entropy value calculated by the anonymiser; the $\log_2 m$ represents the maximum entropy value obtained as the log_2 of the maximum popularity that matches the number of users m. Instead,

$$Pr[E(R, \sigma_{\hat{P}}(\hat{K})) \leq E(R)] = Pr[\mathbb{Y} \leq y] = \int_{0}^{\mathbf{x}''} F(\mathbf{s})\, \mathrm{d}\mathbf{s} \tag{2}$$

captures the likelihood of the anonymiser failing in privacy guarantees.

To better clarify Eqs. 1 and 2 let us consider the following numerical example.

A Numerical Example. We consider the adversarial knowledge, $ADV^{\hat{K}}$, coinciding with the ADV^T as in [24, 25]. We assume the user sets the entropy value (that is the privacy level) to 1.48, represented by the straight line parallel to x-axis in Fig. 4. We also assume three users being visiting the region for a total of 48 footprints, while the ADV^T knowledge is split in two time frames: $\hat{T}_1 = morning$ and $\hat{T}_2 = afternoon$. We use the fixed entropy value (as the one that would be considered by the solution in [40])

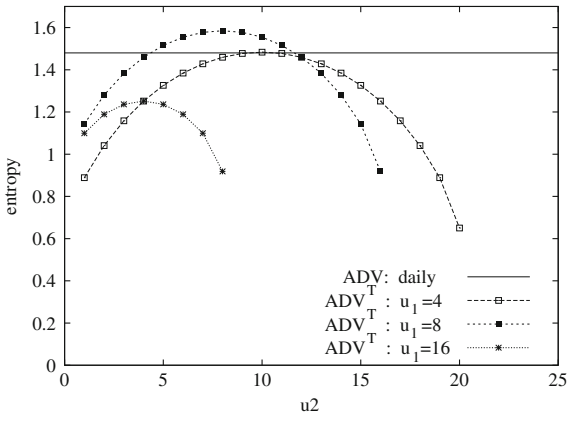


Fig. 4 Comparing entropy between ADV and ADV^T: $\hat{T}_2$ (*afternoon*) footprints distribution, $u_{1,\hat{T}_2} = 4, 8, 16$

to compare with different $ADV^{\hat{K}}$ footprints distributions, sampled as possible $ADV^{\hat{K}}$ knowledge at time frame $\hat{T}_2 = afternoon$. The different scenarios for footprints in $\hat{T}_2$ are obtained as follows: (i) we fix the subset of total ADV footprints for the time frame $\hat{T}_2$, 24 out of 48 in our example; (ii) we fix the number of footprints for user $u_{1,\hat{T}_2}$; (iii) we let $u_{2,\hat{T}_2}$ vary (x-axis), $u_{3,\hat{T}_2}$ being determined once u_1 and u_2 are known. We report the entropy values computed for $u_{1,\hat{T}_2}$, $u_{2,\hat{T}_2}$, and $u_{3,\hat{T}_2}$ on the y-axis. The analytical results computed on these example scenarios are reported in Fig. 4. The results confirm the claim of Theorem 1—the actual level of entropy for ADV^T can be smaller or greater than the one expected for ADV.

In Fig. 4 three curves are plotted for ADV^T, setting respectively $u_{1,\hat{T}_2} = 4$, $u_{1,\hat{T}_2} = 8$, and $u_{1,\hat{T}_2} = 16$. Consistently with Theorem 1, varying footprints distributions may result in ADV^T entropy values (thus adversary effort) much lower than the one calculated for ADV. This is the case for the two curves in Fig. 4 obtained with $u_{1,\hat{T}_2} = 4$ and $u_{1,\hat{T}_2} = 16$. ADV^T entropy values greater than 1.48 (see Fig. 4, ADV^T curve $u_{1,\hat{T}_2} = 8$) raise another issue. Indeed, on the one hand a greater entropy for ADV^T (compared to the one for ADV) might imply a privacy level higher than the one requested. On the other hand this implies a loss in the service accuracy—cloaking the user in an area bigger than the necessary. While we plotted only the results for the entropy, the curves we computed for popularity reflect a shape similar to the ones for entropy—popularity curves have the maximum value of 3 for the uniform distribution obtained setting $u_{1,\hat{T}_2} = 8$, $u_{1,\hat{T}_2} = 8$, and $u_{1,\hat{T}_2} = 8$.

Theorem 1 proves that the problem related to considering context features in designing privacy solutions is actual. However, one might wonder how much likely is that the distributions of footprints falls in the case of Theorem 1. In fact, if the chances to fall into such a scenario were very small, this could not be considered a big concern. In the following, we show that the chances to match the conditions requested for Theorem 1 to hold are not negligible.

To investigate this aspect we considered the following example. In a scenario with two users, we set the number of footprints for the two users to $u_1 = 5$ and $u_1 = 8$, respectively. We vary all the possible distributions of the user footprints split into two time frames $\hat{T}_1 = morning$ and $\hat{T}_2 = afternoon$. For each possible distribution we calculate the corresponding entropy. Assuming each distribution to be equally probable, we thus calculate the ratio between the number of occurrences of each entropy value obtained and the total number of possible distributions, 54 in our example. The resulting probability density function is shown in Fig. 5. In particular, Fig. 5 reports on the probability density of the observed entropy. The entropy calculated for the total number of user footprints is 0.96. It is represented by a vertical line to highlight the points closest to this value. Small squares represent the relation between entropy values (x-axis) and their corresponding probability density (y-axis). We can also observe that the highest probability (0.26) is reached for the entropy value zero obtained for all the distributions, in which at least one of the two users has zero footprints—14 cases in our example. Figure 6 reports the entropy values obtained for each footprints distribution considered at time frame $\hat{T}_1 = morning$. On the x-axis we vary the footprints value for user $u_{1,\hat{T}_1}$, on the

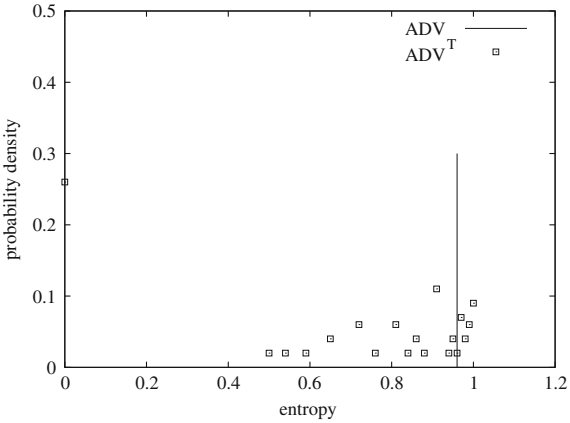


Fig. 5 ADV^T entropy: probability density function ($u_1 = 5, u_2 = 8$)

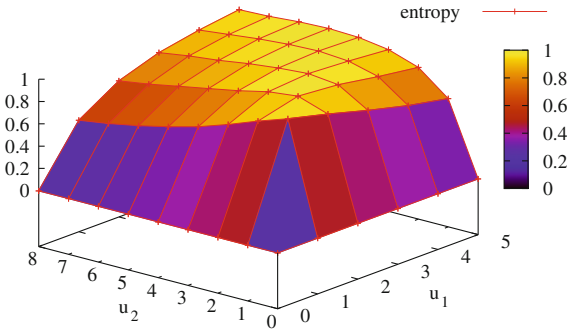


Fig. 6 ADV^T entropy: $\hat{T}_1$ (*morning*) footprints distributions ($u_1 = 5, u_2 = 8$)

y-axis the ones for user $u_{2,\hat{T}_1}$, and on the z-axis we show the resulting entropy. We notice that the values for $u_{2,\hat{T}_2}$ and $u_{2,\hat{T}_2}$ can be derived, once determined the value for $u_{1,\hat{T}_1}$ and $u_{2,\hat{T}_1}$, leveraging the above assumptions on the total number of footprints per user. From Fig. 6 we can observe that the maximum entropy is obtained, as expected, when the numbers of footprints for user u_1 and user u_2 are the same. We can observe this in the diagonal that goes from point ($u_1 = 0, u_2 = 0$) to the point ($u_1 = 5, u_2 = 5$). From this diagonal, when the values for u_2 remains in the high range (e.g. $u_2 = 8$), the entropy remains high. However, when one of the two values decreases, the entropy decreases accordingly. In particular, as already noticed, when one of the two values is equal to zero, the entropy also goes to zero.

5 Experimenting with Real Data

The aim of this section is to discuss the results obtained investigating a real scenario. In particular, we considered an existing data set of footprints information. The series of experiments using real data confirm the observation that the footprints based

models either does not provide the intended privacy, or decreases the quality of LBSs when facing a realistic adversary such as $ADV^{\hat{K}}$.

The *GeoLife GPS trajectories data set* [26] is provided by Microsoft Research Asia [41–43]. It contains traces of 165 users, collected over a two years period: from April 2007 to August 2009. A portion of the data set, composed of 32 users, also records users outdoor movements. The considered activities include everyday life activities like going to work and going back home, as well as those related to sport and entertainment. As a result, each trajectory has a set of transportation mode labels indicating whether a user is driving, taking a bus, riding a bike or walking. Each GPS trajectory in the data set takes the form of a sequence of time-stamped points, $(timestamp, id, p)$, where $p = (x, y)$ is the location (latitude, longitude) of the user (GPS logged) identified by *id* at time *timestamp*; associated to the user *id* are a set of transportation mode labels, one for each of her trajectories. We transform the latitude and longitude coordinates (x, y) provided by the data set in the UTM (Universal Transverse of Mercator) system obtaining a grid-based representation for locations.

For the analysis, we considered the region that delimits the Beijing city area (referred as R_B) and October 2008 as time period. Figure 7 reports an overall view of the footprints in the data set for this region and this period of time. In particular, each colour represent a user and the dots forming the lines represent their geographic location recorded in the data set.

Figures 8, 9, 10 report the footprints distributions when considering different mode of transportation. In particular, we considered users biking (Fig. 8), walking (Fig. 9) and taking a bus (Fig. 10). On the x-axis we represent the user id; on the y-axis, we vary time frames—starting from Monday (Mon) to Sunday (Sun). On the z-axis we

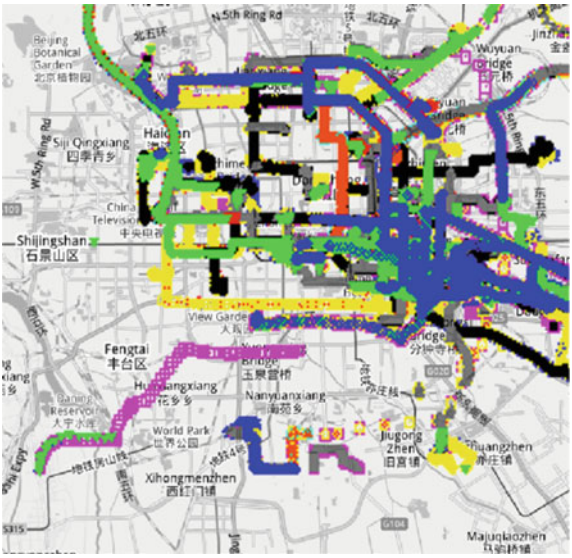


Fig. 7 R_B: Beijing global data set view (Oct 2008)

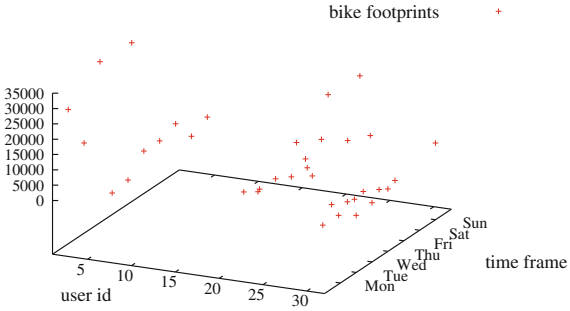


Fig. 8 biking users footprints per time frames

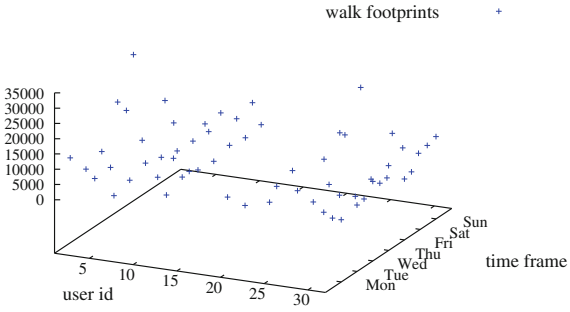


Fig. 9 R_B data set view (Oct 2008): walking users footprints per time frames

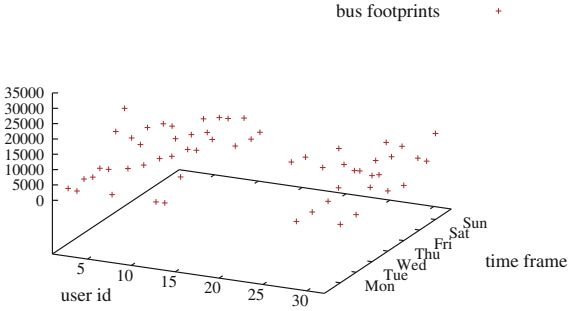


Fig. 10 R_B data set view (Oct 2008): bus users footprints per time frames

show the corresponding number of footprints for each user, in each time frame with the respective transportation mode. We can observe how the walking users in Fig. 9 exhibit the most distributed footprints among users and time-frames. The balanced distribution is likely due to the walking being the primary mobility means. Instead, biking users (Fig. 8) have a footprints distribution that is balanced among time-frames but not among users. We can thus observe how footprints distribution is influenced by the restrictions on data—on biking or walking users dimensions in this case. Looking at Fig. 8, we can notice how there are users (identified by id 1, 6, 27, 30, 31, 32) using the bike daily—while other users (e.g. identified by id from 10 to 18) are not. Hence, we can notice how even human habits affect footprints distribution. Bus users (Fig. 10) show a distributed footprints trend, even if less sparse than the walkers one. In fact, the number of bus users footprints varies from a minimum of 46 to a maximum of 19260 while the walkers footprints fall in the range between 22 and

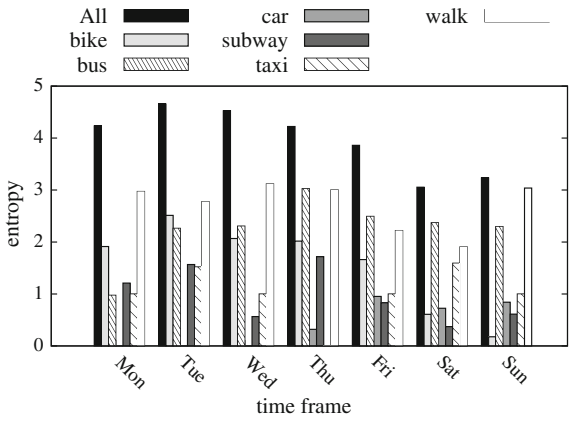


Fig. 11 R_B data set view (Oct 2008): mode of transport footprints, per time frames

52243. The different footprints distributions, produced by the restriction operations on the global data, impact the entropy privacy metric. The impact is depicted in Figs. 11 and 12—fixing one of the two dimensions and letting the other one varying. In particular, in Fig. 11, on the *x*-axis we represent the time frames—being the days of a week, while on the *y*-axis we report the entropy values for each means of transport, for each day of the week.

The highest points represent the entropy calculated considering all the transportation means—i.e. collapsing the mode of transportation dimension. We consider these points as a reference to observe how both the single values and the global trend of the entropy of different transportation mode significantly differs from each other and from the reference points. As a consequence the privacy guarantees depend on the particular restriction on the footprints data considered: that is, the logical predicate used in the restriction operation. As an example, let us consider the *bus* and the *walk* cases. On Mondays, entropy is 0.97 for bus, 2.97 for walk and 4.23 globally. In this example, privacy guarantees are much lower for bus than for walk and far from the LDS value 4.23. On Saturdays, entropy is 2.37 for bus, 1.90 for walk and 3.05 globally. Opposite to Mondays, the privacy guarantees are better for bus than for walk and both are closer to the global value 3.05.

Another view of the footprints data is proposed in Fig. 12. On the *x*-axis we consider the set of transportation labels; on the *y*-axis we plot the corresponding entropy values per day of the week. Here, the reference points are calculated collapsing the time dimension and considering footprints over all the period length (October 2008). From this view on data we can observe that walkers footprints distribution induce the highest entropy, 3.11, on Wednesdays. Also, this value is greater than the value, 2.90, calculated on the whole period. Let us consider the *bus* and the *walk* cases again. On Mondays, entropy is 0.97 for bus and 2.97 for walk. Instead, the entropy values calculated on all period of time is 3.01 for bus and 2.90 for walk. As we can notice, the values are close for walker while are very far from bus users.

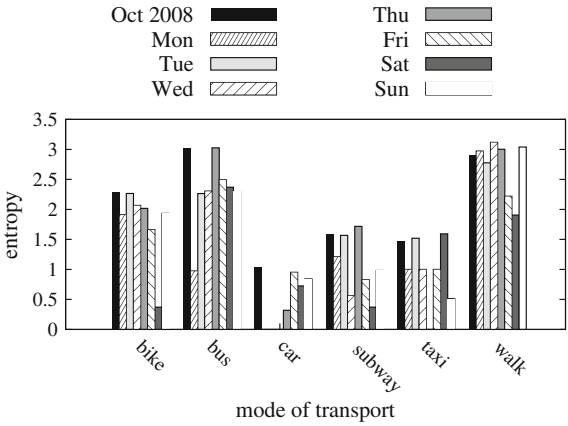


Fig. 12 R_B data set view (Oct 2008): users footprints time frames, per mode of transport

6 Conclusion

In this chapter, we showed that an adversary with a knowledge different from the one used by the anonymiser poses a serious threat to the privacy of users of Location Based Services (LBSs). In particular, we showed that, once different features are taken into consideration, the privacy assurance provided by a state of the art solution does not hold anymore, even when the adversary knowledge about footprints is just partial compared to the one of the anonymiser. We supported our claim with analysis, a simple concrete example and with a thorough study on a real data set. In particular, we considered real mobility traces of GPS users in Beijing, China. The analysis of this data set confirmed our claim on a real user mobility scenario. Also, it showed that the relevance of the highlighted problem is all but negligible. In practical scenarios, the distance between the expected (claimed) privacy level is far away from the one actually granted by the system. We concluded the chapter highlighting which properties must hold for both the anonymiser and the adversary knowledge, in order to guarantee an effective level of user privacy.

Acknowledgments Mauro Conti is supported by a Marie Curie Fellowship funded by the European Commission under the agreement n. PCIG11-GA-2012-321980. This work has been partially supported by the TENACE PRIN Project 20103P34XC funded by the Italian MIUR.

References

1. Ardagna, C., Cremonini, M., De Capitani di Vimercati S., Samarati, P.: An obfuscation-based approach for protecting location privacy. IEEE Trans. Dependable Secure Comput. **8**(1),13–27 (2011)
2. Balsa, E., Troncoso, C., Díaz, C.: Ob-pws: obfuscation-based private web search. In: IEEE Symposium on Security and Privacy, pp. 491–505 (2012)

3. Beresford, A.R., Stajano, F.: Location privacy in pervasive computing. IEEE Pervasive Comput. **2**(1), 46–55 (2003)
4. Bettini, C., Wang, X.S., Jajodia, S.: Protecting privacy against location-based personal identification. In: Proceedings of the 2nd VLDB Workshop on Secure Data Management, pp. 185–199 (2005)
5. Chow, C., Mokbel, M.F., Liu, X.: A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In: GIS '06: Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems, pp. 171–178 (2006)
6. Damiani, M.L., Bertino, E., Silvestri, C.: The probe framework for the personalized cloaking of private locations. Trans. Data Priv. **3**(2), 123–148 (2010)
7. Domingo-Ferrer, J.: k-anonymity. In: Liu, L., Özsu, M.T., (eds.) Encyclopedia of Database Systems, p. 1585. Springer, US (2009). doi:10.1007/978-0-387-39940-9_1503. http://dx.doi.org/10.1007/978-0-387-39940-9_1503
8. Electronic toll collection california (USA). http://www.bayareafastrak.org
9. Freudiger, J., Manshaei M.H., Hubaux J., Parkes, D.C.: On non-cooperative location privacy: a game-theoretic analysis. In: CCS '09: Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 324–337 (2009)
10. Freudiger, J., Manshaei, M.H., Le Boudec, J., Hubaux, J.: On the age of pseudonyms in mobile ad hoc networks. In: INFOCOM: '10: Proceedings of the 29th IEEE International Conference on Computer Communications, pp. 1577–1585 (2010)
11. Freudiger, J., Raya M., Felegyhazi, M., Papadimitratos, P., Hubaux, J.: Mix-zones for location privacy in vehicular networks. In: Win-ITS '07: Proceedings of the First International Workshop on Wireless Networking for Intelligent Transportation Systems (2007).
12. Gedik, B., Liu, L.: A customizable k-anonymity model for protecting location privacy. In: ICDCS '05: Proceedings of the 25th International Conference on Distributed Computing Systems, pp. 620–629 (2005)
13. Ghinita, G., Kalnis, P., Khoshgozaran, A, Shahabi, C., Tan, K.: Private queries in location based services: anonymizers are not necessary. In: SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international Conference on Management of Data, pp. 121–132 (2008)
14. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: MobiSys '03: Proceedings of the 1st International Conference on Mobile systems, Applications and Services, pp. 31–42 (2003)
15. Gruteser, M., Liu, X.: Protecting privacy in continuous location-tracking applications. IEEE Secur. Priv. **2**(2), 28–34 (2004)
16. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc., San Francisco (2006)
17. Hoh, B., Gruteser, M.: Protecting location privacy through path confusion. In: SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks, pp. 194–205 (2005)
18. Hoh, B., Gruteser, M., Xiong, H., Alrabady, A.: Preserving privacy in gps traces via uncertainty-aware path cloaking. In: CCS '07: Proceedings of the 14th ACM Conference on Computer and Communications Security, pp. 161–171 (2007)
19. Kido, H., Yanagisawa, Y., Satoh, T.: An anonymous communication technique using dummies for location-based services. In: ICPS '05: Proceedings of IEEE International Conference on Pervasive Services, pp. 88–97 (2005)
20. Kirmse, A., Udeshi, T., Bellver, P., Shuma, J.: Extracting patterns from location history. In: ACM SIGSPATIAL GIS 2011, pp. 397–400. http://www.sigspatial.org/ (2011)
21. Krumm, J.: A survey of computational location privacy. Pers. Ubiquitous Comput. **13**(6), 391–399 (2009)
22. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Incognito: efficient full-domain k-anonymity. In: SIGMOD '05: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 49–60 (2005)
23. London congestion charge. http://www.tfl.gov.uk/roadusers/

24. Marconi, L., Di Pietro, R., Crispo, B., Conti, M.: Time in privacy preserving LBSs: An overlooked dimension. Int. J. Veh. Technol. **2011**, article ID: 486975, 1–12 (2011)
25. Marconi, L., Di Pietro, R., Crispo, B., Conti, M.: Time warp: how time affects privacy in LBSs. In: ICICS '10: Proceedings of the Twelfth International Conference on Information and Communications Security, pp. 325–339 (2010)
26. Microsoft: Geolife—building social networks using human location history. http://research.microsoft.com/en-us/projects/geolife/ (2008)
27. Mokbel, M.F., Chow, C., Aref, W. G.: The new casper: query processing for location services without compromising privacy. In: VLDB '06: Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 763–774 (2006)
28. Rebollo-Monedero, D., Forné, J., Solanas, A., Martínez-Ballesté, A.: Private location-based information retrieval through user collaboration. Comput. Commun. **33**(6), 762–774 (2010)
29. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In: Proceedings of the IEEE Symposium on Research in Security and Privacy (1998)
30. Schüessler, N., Axhausen, K.W.: Identifying trips and activities and their characteristics from GPS raw data without further information. ETH, Eidgenössische Technische Hochschule Zürich, IVT (2008). http://dx.doi.org/10.3929/ethz-a-005589980
31. Serjantov, A., Danezis, G.: Towards an information theoretic metric for anonymity. In: PET' 02: Proceedings of Privacy Enhancing Technologies Workshop, pp. 41–53 (2002)
32. Shokri, R., Freudiger, J., Hubaux, J.: Unified framework for location privacy. In: PETS '10: Proceedings of the 10th Privacy Enhancing Technologies Symposium, pp. 203–214 (2010)
33. Shokri, R., Troncoso, C., Díaz, C., Freudiger, J., Hubaux, J.-P.: Unraveling an old cloak: k-anonymity for location privacy. In: WPES '10: Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society, pp. 115–118, New York, NY, USA (2010)
34. Solanas, A., Di Pietro, R.:. A linear-time multivariate micro-aggregation for privacy protection in uniform very large data sets. In: MDAI '08: Proceedings of the 5th International Conference on Modeling Decisions for Artificial Intelligence, pp. 203–214 (2008)
35. Solanas, A., Martínez-Ballesté, A.: A ttp-free protocol for location privacy in location-based services. Comput. Commun. **31**(6), 1181–1191 (2008)
36. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl. Syst. **5**(10), 557–570 (2002)
37. Thomas, H., Datta, A.: A conceptual model and algebra for on-line analytical processing in decision support databases. Inf. Syst. Res. **1**(12), 83–102 (2001)
38. Xu, T., Cai, Y.: Location anonymity in continuous location-based services. In: GIS '07: Proceedings of the 15th Annual ACM International Symposium On Advances in Geographic Information Systems, pp. 1–8 (2007)
39. Xu, T., Cai, Y.: Exploring historical location data for anonimity preservation in location-based services. In: INFOCOM 2008: Proceedings of the 27th IEEE Conference on Computer Communications, pp. 547–555 (2008)
40. Xu, T., Cai, Y.: Feeling-based location privacy protection for location-based services. In: CCS' 09: Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 348–357 (2009)
41. Zheng, Y., Chen, Y., Li, Q., Xie, Ma, W.-Y., Xing, X.: Understanding trasportation modes based on gps data for web applications. ACM Trans. Web **4**, 1–36 (2010)
42. Zheng, Y., Li, Q., Chen, Y., Xie, X.: Understanding mobility based on gps data. In: UbiComp 2008: Proceedings of ACM International Conference on Ubiquitous Computing, pp. 312–321 (2008)
43. Zheng, Y., Li, Q., Wang, L., Xie, X.: Learning trasportation modes from raw gps data for geographic application on the web. In: WWW 2008: Proceedings of the 17th International Conference on World Wide Web, pp. 247–256 (2008)

Privacy in Spatio-Temporal Databases: A Microaggregation-Based Approach

Rolando Trujillo-Rasua and Josep Domingo-Ferrer

Abstract Technologies able to track moving objects such as GPS, GSM, and RFID, have been well-adopted worldwide since the end of the 20th century. As a result, companies and governments manage and control huge spatio-temporal databases, whose publication could lead to previously unknown knowledge such as human behaviour patterns or new road traffic trends (e.g., through Data Mining). Aimed at properly balancing data utility with users' privacy rights, several microaggregation-based methods for publishing movement data have been proposed. These methods are reviewed in this book chapter. We highlight challenges in the three stages of the microaggregation process namely, clustering, obfuscation, and privacy and utility evaluation. We also address some of these challenges by presenting yet another microaggregation-based method for privacy-preserving publication of spatio-temporal databases.

1 Introduction

The already mature establishment of telecommunication and wireless technologies has impulsed the collection of spatio-temporal data at a large scale. To fully exploit the analytical usefulness of these data, they eventually need to be released to researchers and/or analysts. Doing so, useful knowledge can be acquired and applied to, for example, intelligent transportation, traffic monitoring, urban and road planning, etc.

R. Trujillo-Rasua (✉)
Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Walferdange, Luxembourg
e-mail: rolandotrujillo@uni.lu

J. Domingo-Ferrer
Department of Computer Engineering and Mathematics, Universitat Rovira i Virgili, Catalonia, Spain
e-mail: josepdomingo@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*, Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_11

However, spatio-temporal data in form of individuals' trajectories are likely to contain sensitive information that users expect to keep private. Consequently, the publication or the outsourcing of databases of trajectories should properly balance data utility with users' privacy rights.

While data utility preservation solely depends on the data, privacy protection needs to consider, in addition, the potential of the adversary. The adversary capability is normally defined as background knowledge learned from other public source of information (e.g., census data or social networks). Knowing the times at which an individual visited a few locations can help an adversary to identify the individual's trajectory in the published database, and therefore learn the individual's other locations at other times. All this makes simple de-identification realized by removing identifying attributes a naive protection mechanism. Hence, more sophisticated privacy-preserving techniques ought to be considered.

Contributions. In this book chapter we review the literature on microaggregation-based methods for privacy-preserving trajectory data publication. In particular, we focus on similarity measures for clustering trajectories and privacy models based on k-anonymity. Amongst those privacy models, we concentrate in (k, δ)-anonymity [5, 6] and prove that it does not preserve privacy in the sense of k-anonymity for $\delta > 0$. We also present a distance between trajectories able to compare trajectories that are not defined over the same time span. Based on this distance, a microaggregation-based approach that preserves original locations (i.e, contain no fake, perturbed or generalized location) is proposed and empirically evaluated by using a real-life dataset.

Organization. Section 2 reviews the k-anonymity concept applied to the trajectory anonymization problem and describes expected properties of the similarity measure used for microaggregation. A flaw in the (k, δ)-anonymity concept is shown in Sect. 3. Our method and distance between trajectories are presented in Sect. 4, which are empirically evaluated in Sect. 5. Section 6 summarizes and concludes the book chapter.

2 Related Work

Samarati and Sweeney [1] proposed in 1998 a novel privacy model named k-anonymity. K-anonymity is based on the concept of *quasi-identifiers*, which are defined as any set of attributes that can potentially appear in publicly available datasets that contain identifiers. A database is said to satisfy k-anonymity if each combination of values of quasi-identifier attributes is shared by at least k records. Therefore, k-anonymity ensures that an adversary (even provided with background knowledge) cannot pinpoint the identity behind a record with probability higher than $1/k$.

A popular and effective technique to achieve k-anonymity is microaggregation [2]. The microaggregation technique works in two stages:

1. *Clustering*. The original records are partitioned into clusters based on some similarity measure. Each cluster contains at least k records and typically no more than $2k - 1$ [3].
2. *Obfuscation*. Each cluster is anonymized individually by obfuscation. The obfuscation may be based on an aggregation operator like the average or the median, or can also be achieved by replacing the records in the cluster with synthetic or partially synthetic data.

In 2006, microaggregation was proposed for location k-anonymity in location-based services [4], but achieving k-anonymity using microaggregation in spatio-temporal data is not straightforward. In a trajectory, any location can be regarded as a quasi-identifier attribute [5]. In this case, k-anonymity would require each anonymized trajectory to be equal to, at least, $k - 1$ other anonymized trajectories. This undoubtedly causes a huge information loss.

To overcome this issue, several trajectory similarity measures and ad-hoc privacy models based on k-anonymity have been proposed [5–9, 11–13]. Both aspects of the microaggregation process are discussed in detail next.

2.1 Distances Between Trajectories

In microaggregation, selecting the *best* distance is of paramount importance. However, what does *best* mean in the context of spatio-temporal data publication could have different, and sometimes contradictory, answers. For instance, some applications (e.g., urban traffic monitoring) might need precise temporal information, whilst others (e.g., evaluation of touristic places attractiveness) deal well with coarse-grained temporal data. We thus list next a few desirable properties of a distance measure for trajectories.

2.1.1 Uncertain Sampling Rate

Trajectories can be recorded at different sampling rate either due to performance issues or technology singularity. The difference in the sampling rate, which typically lead to differences in the size of the trajectories, should has no effect on the result of the distance measure. Neither the Euclidean-based distances used in [5, 7, 8] nor the EDR or the Log-cost distances adopted in [6, 9], respectively, meet this property.

2.1.2 Noise Resiliancy

Several outlier detection mechanism for spatio-temporal data exist. However, subtle differences might appear when comparing two trajectories, which could be regarded as a kind of “noise”, but definitely not as outliers. See Fig. 1 for an example. There,

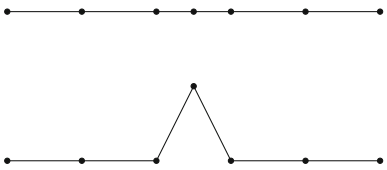


Fig. 1 Two trajectories that are equal except in the peak. They are represented in different planes for visualization purpose only

two identical (except in one location) trajectories are shown. However, distance measures, such as the Frechet distance [10], do not deal well with this scenario. Others, such as the EDR distance, has mechanisms to ignore this "noise" and would consider both trajectories to be equal.

2.1.3 Shape Preservation

The *flow* of the two curves (trajectories) need also to be taken into account. Said differently, a trajectory should not be treated as a set of locations (e.g., see the Hausdorff distance) but as a sequence of locations.

2.1.4 Other Properties

(i) Combine the spatial and the time dimensions (e.g., [7, 8]). (ii) Meet the triangle inequality (e.g., the Euclidean distance). (iii) Have low computational complexity (the Frechet distance is an example of a computationally expensive distance).

In Sect. 4.1 we present our own similarity measure specifically designed for clustering trajectories that might not overlap in time.

2.2 Privacy Models

Privacy models for trajectory anonymization heavily depend on the assumptions about the data and the adversary's knowledge. A trajectory might be downgraded to a location sequence (e.g., as in [12]), which simplifies the model by removing the time dimension from the problem. Other approaches assume that the data owner anonymizing the database knows the set of quasi-identifiers used by the adversary. Consequently, those parts of the trajectories matching the adversary knowledge are simply removed from the published data [11].

A conservative, yet common, assumption is that every location could be regarded as a quasi-identifier. This models then define privacy as the highest re-identification probability for all the users in the dataset. In order to achieve k-anonymity under this assumption, the obfuscation method should transform the trajectories in a cluster in such a way they become indistinguishable. In this regard, different obfuscation

methods for trajectory anonymization have been proposed (e.g., generalization [9, 12, 13], spatial translation [5, 6], and permutation [7, 8].)

In 2008, the (k, δ)-anonymity concept [5], which exploits the spatial uncertainty in the trajectory recording process, was proposed. The parameter k has the same meaning as in k-anonymity, while δ is a lower bound of the uncertainty radius when recording locations. We show in the next section that, for any $\delta > 0$ (that is, whenever there is actual uncertainty), (k, δ)-anonymity does not offer trajectory k-anonymity.[1] As a result, the anonymization methods *Never Walk Alone* (NWA, [5]) and *Wait for Me* (W4M, [6]) preserve the claimed user privacy when $\delta = 0$ only.

3 Privacy Analysis of (k, δ)-Anonymity

The (k, δ)-anonymity privacy notion is based on the assumption that trajectories are imprecise by nature. Unlike records in traditional databases, trajectory data do not remain constant over time, because a moving object should report its position in real-time. However, this is impractical due to performance and wireless-bandwidth overhead. For this reason, Trajcevski et al. [14] suggest that a moving object and the server should reach an agreement consisting on an uncertainty threshold δ, meaning that a position is reported only when it deviates from its expected location by δ or more. Considering so, a moving object does not draw a trajectory anymore, but an uncertain trajectory defined by a trajectory τ and an uncertainty threshold δ.

Definition 1 (*Trajectory*) A *trajectory* is an ordered set of time-stamped locations

$$\tau = \{(t_1, x_1, y_1), \ldots, (t_n, x_n, y_n)\} \ ,$$

where $t_i < t_{i+1}$ for all $1 \leq i < n$.

Notation. For any time-stamp $t_1 \leq t \leq t_n$, the function $\tau(t)$ outputs the location of τ at time t. If $t = t_i$ for some $i \in \{1, \cdots, n\}$ then $\tau(t) = (x_i, y_i)$, otherwise $\tau(t)$ is the linear interpolation of the poly-line τ at time t. Similarly, $\tau(t)[x]$ and $\tau(t)[y]$ denote the spatial coordinates of the location $\tau(t)$.

Definition 2 (*Uncertain trajectory*) An uncertain trajectory is a pair (τ, δ) where τ is a trajectory and δ is an uncertainty threshold. Geometrically, the uncertain trajectory is defined as the locus

$$UT(\tau, \delta) = \{(t, x, y) | d((x, y), (\tau(t)[x], \tau(t)[y])) \leq \delta\} \ ,$$

where $d((x_1, y_1), (x_2, y_2))$ represents the Euclidean distance between the locations (x_1, y_1) and (x_2, y_2).

[1] The proof and analysis provided in Sect. 3 can also be found in the original paper [15].

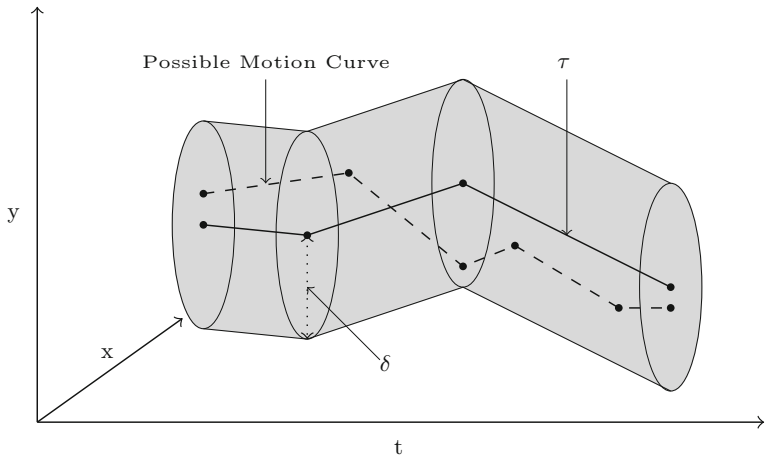


Fig. 2 A trajectory τ and its uncertain trajectory $UT(\tau, \delta)$. A possible motion curve within $UT(\tau, \delta)$ is also shown

As shown in Fig. 2, an uncertain trajectory $UT(\tau, \delta)$ is the union of all the cylinders of radius δ centered in the lines formed by (x_i, y_i) and (x_{i+1}, y_{i+1}) for every $1 \leq i < n$. Then, any continuous function $PMC^{\tau} : [t_1, t_n] \rightarrow \mathbb{R}^2$ such that $PMC^{\tau}([t_1, t_n]) \subset UT(\tau, \delta)$ is said to be a *possible motion curve* of the uncertain trajectory $UT(\tau, \delta)$.

If a trajectory τ_1 is a possible motion curve of the uncertain version (τ_2, δ) of another trajectory τ_2 and viceversa (τ_2 is a possible motion curve of (τ_1, δ)), then τ_1 and τ_2 are said to be *co-localized* with respect to δ [5, 6]. This relation is denoted as $Coloc_{\delta}(\tau_1, \tau_2)$ and provides the rationale behind (k, δ)-anonymity.

Definition 3 ((k, δ)-*anonymity set*) Given an uncertainty threshold δ, a set of trajectories S is considered an anonymity set if and only if $Coloc_{\delta}(\tau_i, \tau_j)\ \forall \tau_i, \tau_j \in S$.

Then, (k, δ)-anonymity is defined as follows in [5, 6]:

Definition 4 ((k, δ)-*anonymity*) Given a database of trajectories $\mathcal{D}$, an uncertainty threshold δ, and an anonymity threshold k, (k, δ)-anonymity is satisfied if, for every trajectory $\tau \in \mathcal{D}$, there exists a (k, δ)-anonymity set $S \subseteq \mathcal{D}$ such that $\tau \in S$ and $|S| \geq k$.

In order to evaluate the privacy offered by (k, δ)-anonymity, we should rely in a second definition of trajectory k-anonymity under the same assumptions. We then use a privacy notion similar to the ones adopted in [7, 9, 12], which are less restrictive than (k, δ)-anonymity [5, 6] in the sense that the parameter δ is not required.

Definition 5 (*Trajectory k-anonymity*) Let T^* be an anonymized set of trajectories corresponding to an original set of trajectories T. Let $\Pr_{\tau^*}[\tau|\sigma]$ denote the probability of the adversary's correctly linking the anonymized trajectory $\tau^* \in T^*$ with its corresponding original trajectory $\tau \in T$ given that the adversary's knows a strict

subset σ of the locations of τ. Then T^* satisfies trajectory k-anonymity if $\Pr_{\tau^*}[\tau|\sigma] \leq 1/k$ for every $\tau \in T$ and σ subset of the locations of τ.

In Definition 5 above, the adversary's knowledge is represented as a *sub-trajectory* of an original trajectory, that is, as a subset of the set of time-stamped locations of the original trajectory. This background knowledge representation is appropriate for the trajectory anonymization schemes [7, 9, 12]. However, the uncertainty on the data under (k, δ)-anonymity does not permit to assume that the adversary knows a sub-trajectory in the above sense, except when $\delta = 0$ (no uncertainty). For $\delta > 0$, the adversary at best could know a possible motion curve PMC_τ of a trajectory τ contained in the original database $\mathcal{D}$. In other words, the adversary cannot be sure that her knowledge PMC_τ is exactly what was recorded in $\mathcal{D}$. It should be remarked that the adversary's knowledge was not explicitly defined in [5] or [6]. However, it is required in this book chapter in order to provide formal privacy proofs.

Definition 6 The adversary's knowledge in a database $\mathcal{D}$ of uncertain trajectories is defined as a random possible motion curve PMC_τ of some trajectory $\tau \in \mathcal{D}$.

Definition 6 can be seen the other way round: the adversary is assumed to have the ability to acquire true actual locations about a user, such as home address or visited places, but the locations recorded in the database form a random possible motion curve of the adversary's knowledge due to the location uncertainty δ. Note that *not* considering the recorded trajectory as a random possible motion curve of the true original trajectory contradicts the (k, δ)-anonymity concept.

Theorem 1 *Let $\mathcal{D}$ be a database satisfying (k, δ)-anonymity. In general, $\mathcal{D}$ does not satisfy trajectory k-anonymity for any $\delta > 0$.*

Proof We first give a counterexample which satisfies $(2, \delta)$-anonymity for any $\delta > 0$ but does not satisfy trajectory 2-anonymity; we will then generalize the argument for any k. Let τ_1 and τ_2 be two different but co-localized trajectories w.r.t. δ such that each of them consists of a single location. By the co-localization condition, the time stamp of both locations is the same and the distance d between the spatial coordinates of both locations satisfies $0 < d \leq \delta$.

Let $\mathcal{D}$ be the original dataset containing τ_1 and τ_2 only. Let us provide the adversary with a random possible motion curve PMC_{τ_i} where $i \in_R \{1, 2\}$ is randomly chosen. According to Definition 5, trajectory 2-anonymity is achieved if the adversary cannot guess with probability greater than $\frac{1}{2}$ whether $i = 1$ or $i = 2$.

However, let us consider the following adversarial strategy:

1. The adversary computes $d(PMC_{\tau_i}, \tau_1)$ and $d(PMC_{\tau_i}, \tau_2)$.
2. If $d(PMC_{\tau_i}, \tau_1) < d(PMC_{\tau_i}, \tau_2)$, the adversary's guess $i = 1$; otherwise, the adversary's guess is $i = 2$.

Now we will show that the previous strategy achieves a probability of success greater than $\frac{1}{2}$. To that end, let us compute the probability that $d(PMC_{\tau_1}, \tau_1) \geq d(PMC_{\tau_1}, \tau_2)$ for a random PMC_{τ_1}.

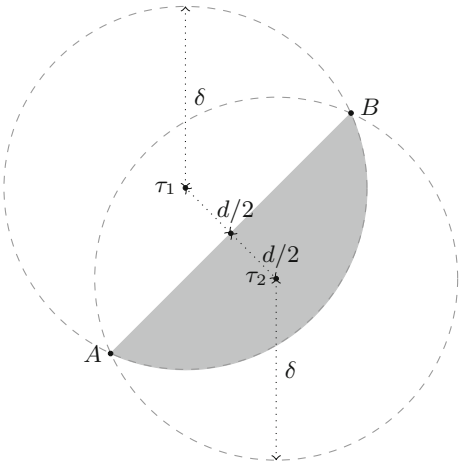


Fig. 3 Two trajectories τ_1 and τ_2 of size 1 such that $d(\tau_1, \tau_2) = d \leq \delta$. The two *circles* that intersect at A and B represent the uncertainty areas of both trajectories according to Definition 2

Let A and B the two points of intersection of the uncertainty circles of τ_1 and τ_2 (see Fig. 3). Then, $d(PMC_{\tau_1}, \tau_1) \geq d(PMC_{\tau_1}, \tau_2)$ only holds when PMC_{τ_1} lies in the arc segment area formed by the points A, B, and the uncertainty circle of τ_1 (shaded area in Fig. 3). Since the line $\overline{AB}$ intersects the line formed by τ_1 and τ_2 in its middle point, it can be concluded that $0 \leq d(A, B) < 2\delta$. As $d(A, B)$ grows towards 2δ, the aforementioned arc segment area becomes asymptotically close to its maximum value $\pi\delta^2/2$. This means that:

$$\Pr(d(PMC_{\tau_1}, \tau_1) \geq d(PMC_{\tau_1}, \tau_2)) < \frac{1}{2}. \tag{1}$$

From Expression (1), it can be concluded that the adversary's success probability is always greater than $\frac{1}{2}$ for any $\delta > 0$, which contradicts 2-anonymity.

The above reasoning can be generalized to any number k of trajectories. The generalized adversarial strategy is:

1. The adversary computes $d(PMC_{\tau_i}, \tau_j)$ for all $j \in \{1, \ldots, k\}$.
2. The adversary's guess is trajectory τ_g such that

$$g = \arg\min_{1 \leq j \leq k} d(PMC_{\tau_i}, \tau_j)$$

By generalizing the geometric argument of Fig. 3, it can be seen that the adversary's success probability with the above strategy is greater than $\frac{1}{k}$. This contradicts trajectory k-anonymity for any k and δ. □

Corollary 1 *The methods NWA* [5] *and W4M* [6] *can only offer trajectory k-anonymity for $\delta = 0$, that is, when all k trajectories in any (k, δ)-anonymity set are identical. In other words, trajectory k-anonymity is offered only when the set of anonymized trajectories consists of clusters containing k or more identical trajectories each.*

4 Our Microaggregation-Based Method

In this section we present an heuristic method, named *SwapLocations*, for privacy-preserving publication of trajectories. SwapLocations is based on microaggregation of trajectories and permutation of locations. It first groups the trajectories into clusters of size at least k based on their similarity and then transforms via location permutation the trajectories inside each cluster to preserve privacy.

For clustering purposes, we present a distance for trajectories which naturally considers both spatial and temporal coordinates. Our distance is able to compare trajectories that are not defined over the same time span, without resorting to time generalization. It can also compare trajectories that are timewise overlapping only partially or not at all.

4.1 Our Similarity Measure

Clustering trajectories requires defining a similarity measure—a distance between two trajectories. Because trajectories are distributed over space and time, a distance that considers both spatial and temporal aspects of trajectories is needed. Many distance measures have been proposed in the past for both trajectories of moving objects and for time series but most of them are ill-suited to compare trajectories for anonymization purposes. Therefore we define a new distance which can compare trajectories that are only partially or not at all timewise overlapping. We believe this is necessary to cluster trajectories for anonymization. We need some preliminary notions.

Definition 7 (*p %-contemporary trajectories*) Two trajectories

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_n^i, x_n^i, y_n^i)\}$$

and

$$T_j = \{(t_1^j, x_1^j, y_1^j), \ldots, (t_m^j, x_m^j, y_m^j)\}$$

are said to be $p\,\%$-contemporary if

$$p = 100 \cdot \min\left(\frac{I}{t_n^i - t_1^i}, \frac{I}{t_m^j - t_1^j}\right)$$

with $I = \max(\min(t_n^i, t_m^j) - \max(t_1^i, t_1^j), 0)$.

Intuitively, two trajectories are 100 %-contemporary if and only if they start at the same time and end at the same time; two trajectories are 0 %-contemporary if and only if they occur during non-overlapping time intervals. Denote the overlap time of two trajectories T_i and T_j as $ot(T_i, T_j)$.

Definition 8 (*Synchronized trajectories*). Given two $p\%$-contemporary trajectories T_i and T_j for some $p > 0$, both *trajectories are said to be synchronized* if they have the same number of locations timestamped within $ot(T_i, T_j)$ and these correspond to the same timestamps. A *set of trajectories is said to be synchronized* if all pairs of $p\%$-contemporary trajectories in it are synchronized, where $p > 0$ may be different for each pair.

If we assume that between two locations of a trajectory, the object is moving along a straight line between the locations at a constant speed, then interpolating new locations is straightforward. Trajectories can be then synchronized in the sense that if one trajectory has a location at time t, then other trajectories defined at that time will also have a (possibly interpolated) location at time t. This transformation guarantees that the set of new locations interpolated in order to synchronize trajectories is of minimum cardinality. Algorithm 1 describes this process. The time complexity of this algorithm is $O(|TS|^2)$ where $|TS|$ is the number of different timestamps in the data set.

Algorithm 1 Trajectory synchronization

Require: $\mathcal{T} = \{T_1, \ldots, T_N\}$ a set of trajectories to be synchronized, where each $T_i \in \mathcal{T}$ is of the form:

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_{n^i}^i, x_{n^i}^i, y_{n^i}^i)\};$$

1: Let $TS = \{t_j^i \mid (t_j^i, x_j^i, y_j^i) \in T_i \ : \ T_i \in \mathcal{T}\}$ be all timestamps from all locations of all trajectories;
2: **for all** $T_i \in \mathcal{T}$ **do**
3: **for all** $ts \in TS$ with $t_1^i < ts < t_{n^i}^i$ **do**
4: **if** location having timestamp ts is not in T_i **then**
5: insert new location to T_i having the timestamp ts and coordinates interpolated from the two timewise-neighboring locations;
6: **end if**
7: **end for**
8: **end for**

Definition 9 (*Distance between trajectories*) Consider a set of synchronized trajectories $\mathcal{T} = \{T_1, \ldots, T_N\}$ where each trajectory is written as

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_{n^i}^i, x_{n^i}^i, y_{n^i}^i)\} \ .$$

The *distance between trajectories* is defined as follows. If $T_i, T_j \in \mathcal{T}$ are $p\%$-contemporary with $p > 0$, then

$$d(T_i, T_j) = \frac{1}{p} \sqrt{\sum_{t_\ell \in ot(T_i, T_j)} \frac{(x_\ell^i - x_\ell^j)^2 + (y_\ell^i - y_\ell^j)^2}{|ot(T_i, T_j)|^2}} \ .$$

If $T_i, T_j \in \mathcal{T}$ are $0\,\%$-contemporary but there is at least one subset of $\mathcal{T}$

$$\mathcal{T}^k(ij) = \{T_1^{ijk}, T_2^{ijk}, \ldots, T_{n^{ijk}}^{ijk}\} \subseteq \mathcal{T}$$

such that $T_1^{ijk} = T_i$, $T_{n^{ijk}}^{ijk} = T_j$ and T_ℓ^{ijk} and $T_{\ell+1}^{ijk}$ are p_ℓ%-contemporary with $p_\ell > 0$ for $\ell = 1$ to $n^{ijk} - 1$, then

$$d(T_i, T_j) = \min_{\mathcal{T}^k(ij)} \left(\sum_{\ell=1}^{n^{ijk}-1} d(T_\ell^{ijk}, T_{\ell+1}^{ijk}) \right)$$

Otherwise $d(T_i, T_j)$ is not defined.

The computation of the distance between every pair of trajectories is not exponential as it could seem from the definition. Polynomial-time computation of a distance graph containing the distances between all pairs of trajectories can be done as follows.

Definition 10 (*Distance graph*) A *distance graph* is a weighted graph where

(i) Nodes represent trajectories,
(ii) two nodes T_i and T_j are adjacent if the corresponding trajectories are p%-contemporary for some $p > 0$, and
(iii) the weight of the edge (T_i, T_j) is the distance between the trajectories T_i and T_j.

Now, given the distance graph for $\mathcal{T} = \{T_1, \ldots, T_N\}$, the distance $d(T_i, T_j)$ for two trajectories is easily computed as the minimum cost path between the nodes T_i and T_j, if such path exists. The inability to compute the distance for all possible trajectories (the last case of Definition 9) naturally splits the distance graph into connected components. The connected component that has the majority of the trajectories must be kept, while the remaining components represent outlier trajectories that are discarded in order to preserve privacy. Finally, given the connected component of the distance graph having the majority of the trajectories of $\mathcal{T}$, the distance $d(T_i, T_j)$ for *any two* trajectories on this connected component is easily computed as the minimum cost path between the nodes T_i and T_j. The minimum cost path between every pair of nodes can be computed using the Floyd-Warshall algorithm with computational cost $O(N^3)$, i.e., in polynomial time.

4.2 The SwapLocations Method

Algorithm 2 describes the process followed by the SwapLocations method in order to anonymize a set of trajectories. First, the set of trajectories is partitioned into several clusters. Then, each cluster is anonymized using the SwapLocations function in Algorithm 3.

We limit ourselves to clustering algorithms which try to minimize the sum of the intra-cluster distances or approximate the minimum and such that the cardinality of each cluster is k, with k an input parameter; if the number of trajectories

is not a multiple of k, one or more clusters must absorb the up to $k-1$ remaining trajectories, hence those clusters will have cardinalities between $k+1$ and $2k-1$. This type of clustering is precisely the one used in microaggregation [3]. The purpose of minimizing the sum of the intra-cluster distances is to obtain clusters as homogeneous as possible, so that the subsequent independent treatment of clusters does not cause much information loss. The purpose of setting k as the cluster size is to fulfill trajectory k-anonymity. We employ any microaggregation heuristic for clustering purposes.

Algorithm 2 Cluster-based trajectory anonymization($\mathcal{T}$, R^t, R^s, k)

Require: (i) $\mathcal{T} = \{T_1, \ldots, T_N\}$ a set of original trajectories such that $d(T_i, T_j)$ is defined for all $T_i, T_j \in \mathcal{T}$, (ii) R^t a time threshold and R^s a space threshold;

1: Use any clustering algorithm to cluster the trajectories of $\mathcal{T}$, while minimizing the sum of intra-cluster distances measured with the distance of Definition 9 and ensuring that minimum cluster size is k;
2: Let $C_1, C_2, \ldots, C_{n_\mathcal{T}}$ be the resulting clusters;
3: **for all** clusters C_i **do**
4: $C_i^\star$ = SwapLocations(C_i, R^t, R^s); // Algorithm 3
5: **end for**
6: Let $\mathcal{T}^\star = C_1^\star \cup \cdots \cup C_{n_\mathcal{T}}^\star$ be the set of anonymized trajectories.

The SwapLocations function (Algorithm 3) begins with a random trajectory T in C. The function attempts to cluster each unswapped triple λ in T with another $k-1$ unswapped triples belonging to different trajectories such that: (i) the timestamps of these triples differ by no more than a time threshold R^t from the timestamp of λ; (ii) the spatial coordinates differ by no more than a space threshold R^s. If no $k-1$ suitable triples can be found that can be clustered with λ, then λ is removed; otherwise, random swaps of triples are performed within the formed cluster. Randomly swapping this cluster of triples guarantees that any of these triples has the same probability of remaining in its original trajectory or becoming a new triple in any of the other $k-1$ trajectories. Note that Algorithm 3 guarantees that every triple λ of every trajectory $T \in C$ will be swapped or removed.

The method SwapLocations meets trajectory k-anonymity in the sense of Definition 5. Refer to the original work [7] for details on the privacy analysis of SwapLocations.

5 Empirical Results

In this section we evaluate the SwapLocations method by using a real-life data set of cab mobility traces that were collected in the city of San Francisco [16][2]. We consider three utility measures: (i) percentage of removed trajectories, (ii) percentage

[2] A more comprehensive empirical evaluation can be found in the original paper where SwapLocations is introduced [7].

Algorithm 3 SwapLocations(C, R^t, R^s)

Require: (i) C a cluster of trajectories to be transformed, (ii) R^t a time threshold and R^s a space threshold;

1: Mark all triples in trajectories in C as "unswapped";
2: Let T be a random trajectory in C;
3: **for all** "unswapped" triples $\lambda = (t_\lambda, x_\lambda, y_\lambda)$ in T **do**
4: Let $U = \{\lambda\}$; // Initializing U with $\{\lambda\}$
5: **for all** trajectories T' in C with $T' \neq T$ **do**
6: Look for an "unswapped" triple $\lambda' = (t_{\lambda'}, x_{\lambda'}, y_{\lambda'})$ in T' minimizing the intra-cluster distance in $U \cup \{\lambda'\}$ and such that:

$$|t_{\lambda'} - t_\lambda| \leq R^t$$

$$0 \leq \sqrt{(x_{\lambda'} - x_\lambda)^2 + (y_{\lambda'} - y_\lambda)^2} \leq R^s \ ;$$

7: **if** λ' exists **then**
8: $U \leftarrow U \cup \{\lambda'\}$;
9: **else**
10: Remove λ from T;
11: Goto line 3 in order to analyze the next triple λ;
12: **end if**
13: **end for**
14: Randomly swap all triples in U;
15: Mark all triples in U as "swapped";
16: **end for**
17: Remove all "unswapped" triples in C;
18: **return** C.

of removed locations, (iii) and spatio-temporal range queries as proposed in [14]. The latter is described in more detail next.

5.1 Spatio-Temporal Range Queries

Trajcevski et al. proposed in [14] six spatio-temporal range queries. For the sake of simplicity, we just keep the two more relevant for our experiments: *Sometime Definitely Inside* (SI) and *Always Definitely Inside* (AI).

- $SI(T, R, t_b, t_e)$ is *true* if and only if there exists a time $t \in [t_b, t_e]$ at which every possible motion curve PMC^T of an uncertain trajectory $U(T, \sigma)$ is inside region R. For a non-uncertain T, the previous condition can be adapted as: if and only if there exists a time $t \in [t_b, t_e]$ at which T is inside R.
- $AI(T, R, t_b, t_e)$ is *true* if and only if at every time $t \in [t_b, t_e]$, every possible motion curve PMC^T of an uncertain trajectory $U(T, \sigma)$ is inside region R. For a non-uncertain T, the previous condition becomes: if and only if at every time $t \in [t_b, t_e]$, trajectory T is inside R.

We accumulate the number of trajectories in a set of trajectories $\mathcal{T}$ that satisfy the SI or AI range queries using the SQL style code below.

– Query $\mathcal{Q}_1(\mathcal{T}, R, t_b, t_e)$:
  `SELECT COUNT (*) FROM` $\mathcal{T}$ `WHERE SI(`$\mathcal{T}$`.traj, R,` t_b, t_e`)`
– Query $\mathcal{Q}_2(\mathcal{T}, R, t_b, t_e)$:
  `SELECT COUNT (*) FROM` $\mathcal{T}$ `WHERE AI(`$\mathcal{T}$`.traj, R,` t_b, t_e`)`

Then, we define two different *range query distortions*:

– $\text{SID}(\mathcal{T}, \mathcal{T}^\star) = \frac{1}{|\xi|} \sum_{\forall <R,t_b,t_e> \in \xi} \frac{|\mathcal{Q}_1(\mathcal{T},R,t_b,t_e) - \mathcal{Q}_1(\mathcal{T}^\star,R,t_b,t_e)|}{\max(\mathcal{Q}_1(\mathcal{T},R,t_b,t_e), \mathcal{Q}_1(\mathcal{T}^\star,R,t_b,t_e))}$ where ξ is a set of SI queries.
– $\text{AID}(\mathcal{T}, \mathcal{T}^\star) = \frac{1}{|\xi|} \sum_{\forall <R,t_b,t_e> \in \xi} \frac{|\mathcal{Q}_2(\mathcal{T},R,t_b,t_e) - \mathcal{Q}_2(\mathcal{T}^\star,R,t_b,t_e)|}{\max(\mathcal{Q}_2(\mathcal{T},R,t_b,t_e), \mathcal{Q}_2(\mathcal{T}^\star,R,t_b,t_e))}$ where ξ is a set of AI queries.

5.2 Results on Real-Life Data

The San Francisco cab data set [16] we used consists of several files each of them containing the GPS information of a specific cab during May 2008. Each line within a file contains the space coordinates (latitude and longitude) of the cab at a given time. However, the mobility trace of a cab during an entire month can hardly be considered a single trajectory. We used big time gaps between two consecutive locations in a cab mobility trace to split that trace into several trajectories.

For our experiments we considered just one day of the entire month given in the real-life data set, but the empirical methodology described below could be extended to several days. In particular, we chose the day between May 25 at 12:04 h and May 26 at 12:04 h because during this 24 h period there was the highest concentration of locations in the data set. We also defined the maximum time gap in a trajectory as 3 min; above 3 min, we assumed that the current trajectory ended and that the next location belonged to a different trajectory. This choice was based on the average time gap between consecutive locations in the data set, which was 88 s; hence, 3 min was roughly twice the average. In this way, we obtained 4,582 trajectories and 94 locations per trajectory on average.

The next step was to filter out trajectories with strange features (outliers). These outliers could be detected based on several aspects like velocity, city topology, etc. We focused on velocity and defined 240 km/h as the maximum speed that could be reached by a cab. Consequently, the distance between two consecutive locations could not be greater than 12 km because the maximum within-trajectory time gap was 3 min. This allowed us to detect and remove trajectories containing obviously erroneous locations; Fig. 4 shows one of these removed outliers where a cab appeared to have jumped far into the sea probably due to some error in recording its GPS coordinates. Altogether, we removed 45 outlier trajectories and we were left with a data set of 4547 trajectories with an average of 93 locations per trajectory. Figure 5 shows the ten longest trajectories (in number of locations) in the final data set that we used.

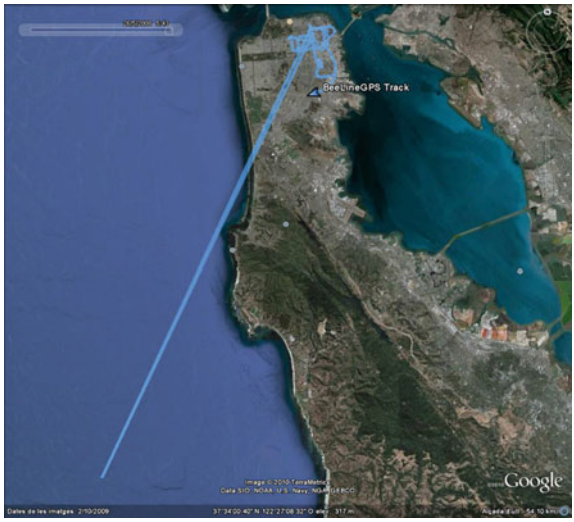


Fig. 4 Example of an outlier trajectory in the original real-life data set

Fig. 5 Ten longest trajectories in the filtered real-life data set

We first consider the percentage of removed trajectories and the percentage of removed locations as utility measures. Table 1 shows how SwapLocations performs in terms of both.

Finally, Table 2 reports the performance of SwapLocations regarding spatio-temporal range queries. We picked random time intervals of length at most 20 min. Also, random uncertain trajectories with uncertainty threshold of size at most 7 km were chosen as the regions, which is roughly a quarter of the average distance of all trajectories. It can be seen that the SwapLocations method provides lower range

Table 1 Percentage of trajectories (columns labeled with **T**) and locations (columns labeled with **L**) removed by SwapLocations for several values of k and several space thresholds R^s on the real-life data set

$R^s \backslash k$	2		4		6		8		10		15	
	T	**L**	**T**	**L**	**T**	**L**	**T**	**L**	**T**	**L**	**T**	**L**
1	23	43	40	64	49	71	58	74	62	77	71	81
2	19	29	34	47	42	54	50	58	54	60	50	66
4	14	17	27	29	35	35	40	40	45	41	54	49
8	9	10	19	19	25	25	31	29	34	31	42	38
16	5	7	11	16	17	22	20	27	23	30	32	38
32	1	7	2	15	3	22	4	27	5	30	8	38
64	0	6	0	15	0	22	0	27	0	30	0	38
128	0	6	0	15	0	22	0	27	0	30	0	38

Percentages have been rounded to integers for compactness

Table 2 Range query distortion caused by SwapLocations in terms of SID (columns labeled with **S**) and AID (columns labeled with **A**), for several values of k and several space thresholds R^s

$R^s \backslash k$	2		4		6		8		10		15	
	S	**A**	**S**	**A**	**S**	**A**	**S**	**A**	**S**	**A**	**S**	**A**
1	13	22	18	27	20	29	19	29	24	31	25	34
2	16	24	25	34	26	35	24	35	27	37	27	37
4	18	25	30	37	33	41	34	42	38	46	38	45
8	21	27	34	40	38	44	40	46	44	50	48	54
16	20	26	36	42	42	47	45	50	50	54	53	58
32	21	26	39	44	45	49	48	53	53	57	58	62
64	20	25	39	44	46	50	51	54	54	57	61	64
128	21	26	39	44	48	50	51	56	54	58	61	64

A range query distortion x is represented as the integer rounding of $x * 100$ for compactness.

query distortion for every value of k when the space threshold is small, i.e. when the total space distortion is also small. However, the smaller the space threshold, the larger the number of removed trajectories and locations (see Table 1). This illustrates the trade-off between the utility properties considered.

6 Conclusions

Several microaggregation-based methods for privacy-preserving spatio-temporal data publication have been proposed up to date. They mostly differ in the similarity measure, the obfuscation method, and the privacy model considered. In this book chapter we highlighted relevant properties for trajectory similarity measures that should be taken into account for microaggregation. We also described different

privacy models based on k-anonymity in terms of the assumptions on the data and the adversary capabilities. In particular, we provided a proof that invalidates the (k, δ)-anonymity concept for $\delta > 0$. Finally, we presented a similarity measure and a microaggregation-based approach that together deal with non-overlapping trajectories and preserve original locations. The method was evaluated by using a real-life dataset of trajectory data.

Acknowledgments The second author is partially supported by the Government of Catalonia through an ICREA Acadèmia Prize. The following partial supports are also gratefully acknowledged: the Spanish Government under projects TIN2011-27076-C03-01 "CO-PRIVACY" and CONSOLIDER INGENIO 2010 CSD2007-00004 "ARES", and the European Commission under FP7 projects "DwB" and "Inter-Trust". The second author is with the UNESCO Chair in Data Privacy, but the views expressed in this paper neither necessarily reflect the position of UNESCO nor commit that organization.

References

1. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, SRI Computer Science Laboratory (1998)
2. Domingo-Ferrer, J., Torra, V.: Ordinal, continuous and heterogenerous k-anonymity through microaggregation. Data Min. Knowl. Discov. **11**(2), 195–212 (2005)
3. Domingo-Ferrer, J., Mateo-Sanz, J.M.: Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans. Knowl. Data Eng. **14**(1), 189–201 (2002)
4. Domingo-Ferrer, J.: Microaggregation for database and location privacy. In: Proceedings of Next Generation Information Technologies and Systems-NGITS'2006, LNCS 4302, Springer, pp. 233–242 (2006)
5. Abul, O., Bonchi, F., Nanni, M.: Never walk alone: uncertainty for anonymity in moving objects databases. In: Proceedings of the IEEE 24th International Conference on Data Engineering, ICDE 2008, Cancun, Mexico, 7–12 Apr 2008, pp. 376–385 (2008)
6. Abul, O., Bonchi, F., Nanni, M.: Anonymization of moving objects databases by clustering and perturbation. Inf. Syst. **35**(8), 884–910 (2010)
7. Domingo-Ferrer, J., Trujillo-Rasua, R.: Microaggregation- and permutation-based anonymization of movement data. Inf. Sci. **208**, 55–80 (2012)
8. Domingo-Ferrer, J., Sramka, M., Trujillo-Rasua, R.: Privacy-preserving publication of trajectories using microaggregation. In: Proceedings of the SIGSPATIAL ACM GIS 2010 International Workshop on Security and Privacy in GIS and LBS, SPRINGL 2010, San Jose, California, USA, 2 Nov 2010. ACM (2010)
9. Nergiz, M.E., Atzori, M., Saygin, Y., Guc, B.: Towards trajectory anonymization: a generalization-based approach. Trans. Data Priv. **2**(1), 47–75 (2009)
10. Alt, H., Godau, M.: Computing the Fréchet distance between two polygonal curves. In: International Journal of Computational Geometry & Applications, vol. 5, pp. 75–91, 1995. http://dblp.uni-trier.de/db/journals/ijcga/ijcga5.html
11. Terrovitis, M., Mamoulis, N.: Privacy preservation in the publication of trajectories. In: Proceedings of the 9th IEEE International Conference on Mobile Data Management, MDM 2008, Beijing, China, 27–30 Apr 2008, pp. 65–72 (2008)
12. Monreale, A., Andrienko, G., Andrienko, N., Giannotti, F., Pedreschi, D., Rinzivillo, S., Wrobel, S.: Movement data anonymity through generalization. Trans. Data Priv. **3**(2), 91–121 (2010)

13. Yarovoy, R., Bonchi, F., Lakshmanan, L.V.S., Wang, W.H.: Anonymizing moving objects: how to hide a mob in a crowd? In: Proceedings of the 12th International Conference on Extending Database Technology, EDBT 2009, Saint Petersburg, Russia, 24–26 March 2009, volume 360 of ACM International Conference Proceeding Series, pp. 72–83. ACM (2009)
14. Trajcevski, G., Ouri, O., Hinrichs, K., Chamberlain, S.: Managing uncertainty in moving objects databases. ACM Trans. Database Syst. **29**(3), 463–507 (2004)
15. Trujillo-Rasua, R., Domingo-Ferrer, J.: On the privacy offered by (k, δ)-anonymity. Inf. Syst. **38**(4), 491–494 (2013)
16. Piorkowski, M., Sarafijanovoc-Djukic, N., Grossglauser, M.: A parsimonious model of mobile partitioned networks with clustering. In: The First International Conference on COMmunication Systems and NETworkS (COMSNETS), Bangalore, India, January (2009)

A Prototype for Anonymizing Trajectories from a Time Series Perspective

Sergi Martínez-Bea

Abstract The evolution and expansion of location tracking technologies such as GPS, RFID, etc. and their integration with handheld devices, created a new trend of services and applications based on location information. However, location data is sensible data that could seriously compromise the privacy of the individuals. There is a large body of research in the area of location privacy, where researchers try to tackle this privacy problem. In this article we describe one of the systems implemented in the ARES project to anonymize trajectories of cars in a prototype, following an approach based on time series.

1 Introduction

In the recent years, there has been a huge growth of the use of smartphones by individuals. The majority of those smartphones are equipped with technologies such as GPS, that allow their geolocalization. Because of that, a new market for services that use location information to provide customized information to the end user appeared. However, the use of this kind of data may harm the privacy of the individuals. In order to deal with this threat, several researchers started working in the area of location privacy.

There is a large body of research in the area of location privacy nowadays. In [1], the authors provided a survey where they give an overview of the state-of-the-art in location privacy back in 2009 from the dual prespective of query privacy and trajectory anonymization. In the same year, the corresponding authors of [2–4] presented different approaches to tackle location privacy. In [2] a location privacy approach is presented which consists of a suitable modification of the queries prior

S. Martínez-Bea (✉)
IIIA, Institut d'Investigació en Intel·ligència Artificial CSIC,
Consejo Superior de Investigaciones Científicas Campus de la UAB,
08193 Bellaterra, Catalonia, Spain
e-mail: smartinez@iiia.csic.es

S. Martínez-Bea
UAB, Universitat Autònoma de Barcelona, Barcelona, Spain

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_12

to be sent to the location-based service. [3] focus on static trajectories, by adopting a generalization-based approach. A different point of view is presented in [4] in which the authors study the factors on which location privacy depends, and propose models for expressing and enforcing privacy preferences for location data. Later on, in 2011, a review of the technical challenges of location privacy is presented in [5], together with suggested directions of research towards a comprehensive privacy-preserving framework. The authors of a more recent work [6], investigated how an individual can achieve the privacy goal that the inclusion of his location history in a statistical database with interesting location mining capability does not substantially increase risk to his privacy.

This paper describes one of the systems implemented in the ARES project to anonymize trajectories of cars in a prototype. The approach followed in this system is based on time series [7].

In Sect. 2 we describe the concepts and definitions used in this article. In Sect. 3 we explain how we protect the trajectories. In Sect. 4 we show an overview of the prototype. Finally, we conclude the article with conclusions.

2 Preliminaries

In this section we describe the concepts and definitions which are required in the chapter.

2.1 Protection Methods

A protection method is an algorithm that performs some operations on a given data set in order to achieve a certain level of privacy. Formally, given a data set X, the protection method transforms it to a protected data set X'. In general, X can be seen as a matrix with n rows (records) and m columns (attributes). The attributes can be classified in two different types: identifiers and quasi-identifiers. Identifier attributes are those that unambiguously identify an individual. A good example of an identifier attribute is the personal id number, as it is unique for each individual. Thus, this number unambiguously identify its owner. The other type of attributes, the quasi-identifiers, are those that can not identify an individual by themselves, but a set of them can. This kind of attributes are divided in confidential and non-confidential depending on whether they contain private information. As example of quasi-identifiers we find the hometown and the medical history. The first one is non-confidential while the second is confidential. The hometown or the medical history by themselves will hardly identify an individual in an unambiguous manner, but if we take both of them into account together with other quasi-identifiers we could come up with a combination capable of unambiguously identify an individual.

In the typical scenario, before releasing a data set X with confidential attributes, a protection method p is applied to X, obtaining a protected data set X'. In the dataset X the identifier attributes will be removed or encrypted $X = X_{nc}||X_c$, confidential

quasi-identifier attributes X_c will not be modified so we have $X'_c = X_c$, and the non-confidential quasi-identifier attributes will be protected using the protection method itself in order to preserve the privacy of the individuals. Thus, we have $X'_{nc} = p(X_{nc})$.

Microaggregation One of the most effective data protection methods is microaggregation. That is because this protection method, when all the attributes are protected at the same time, ensures a property called k-anonymity, which consists of having the dataset divided in groups of at least k undistinguishable elements. To achieve k-anonymity, microaggregation builds small clusters of at least k elements and replace the original values with the centroid of the cluster to wich record belongs to.

However, perturbative protection methods have information loss due to their nature. To solve this, multivariate microaggregation is used but at cost of disclosure risk. Multivariate microaggregation divides the attributes into different blocks and applies basic microaggregation to each of them separately. This may cause that the k records which fall into the same cluster for the first block, may fall into a different one in the second.

Additionally, microaggregation methods try to minimize the total sum of distances between all the elements to be protected and the centroid of the cluster where an element belongs to, in order to keep the information loss as low as possible. In general, the larger the value of k the lower the disclosure risk, but the information loss increase. Thus, k has to be large enough to ensure privacy but not too large to compromise the statistical utility of the protected data. That could be done by findind the optimal multivariate microaggregation, but it has been proven to be an NP-Hard problem. Therefore, heuristic methods have been developed, such as the MDAV (Maximum Distance to Average Vector) algorithm.

Algorithm 1: MDAV

Input: X: original microdata, k: integer
Output: X': protected microdata
1: **while** $|X| > k$ **do**
2: Compute the average record $\bar{x}$ of all records X
3: Consider the most distant record x_r to the average record $\bar{x}$
4: Form a cluster around x_r. The cluster contains x_r together with the $k-1$ closest records to x_r
5: Remove these records from microdata file X
6: **if** $|X| > k$ **then**
7: Find the most distant record x_s from record x_r
8: Form a cluster around x_s. The cluster contains x_s together with the $k-1$ closest records to x_s
9: Remove these records from microdata file X
10: **end if**
11: **end while**
12: Form a cluster with the remaining records

MDAV Microaggregation The Maximum Distance to Average Vector algorithm is an heuristic algorithm for clustering records in a given data set X so that each cluster contains at least k records. The MDAV algorithm can be appreciated in Algorithm 1.

In order to apply the MDAV algorithm to different data types, we need to define a distance and average functions, so we know what the most distant record means, which of the closest records of a given record are, and the average record of a set of records, by establishing a distance measure and an average function for the data type being used.

2.2 Evaluation Measures

An evaluation measure can be seen as a function that provides some insight on how good or bad performs a given protection method. Thus, it is important to define properly the evaluation measures as it will help to determine whether a protection method is good or not. In this scenario, the evaluation measures are information loss and disclosure risk.

Information Loss When applying a protection method to a data set there is always some loss of information. This is due to the nature of the perturbative protection methods, that inflict some perturbations that decrease the utility of the protected data.

The measurement of the loss of information can be performed in two ways. If we know what will be the use that the protected data will have, we can apply the same operations to the original and protected data, and measure the difference between both results so we obtain a value that suggests how many information has been lost in the protection. However, this is not the common scenario because it is hard to know the future use of the protected data. Thus, the information loss measures need to be generic enough to be able to reflect the ammount of information that is being lost for a range of data uses.

Disclosure Risk When protecting any data, the main goal is to preserve the privacy of the individuals. By doing so, as we already pointed out, we have some loss of information and, therefore, it is desirable to keep it as low as possible. However, achieving low information loss implies that the protected data will resemble the original increasing this way the risk of disclosure.

One way to measure the risk of disclosure, is to do what an actual attacker would do in case he gets the original an protected data sets. That would be to try to establish some links between the individuals in both data sets.

3 Protecting Trajectories

In Sect. 2 we explained what a protection method is for general data. In this section, we detail a protection method for trajectories. First of all, a trajectory can be defined as a sequence of places or spots that a given object or person has gone through.

We can define each place or spot with their coordinates x and y and a timestamp indicating when the object or person was there. Formally, we define a trajectory as:

$$\{(x_1, y_1, t_1), (x_2, y_2, t_2), \ldots, (x_n, y_n, t_n)\} \text{ where } n \in \mathbb{N}$$

Given this definition, we now can better explain the functions needed to perform microaggregation to trajectory data. Recall that we need to define a distance function and an average function. As in [7], we define the following distance and average functions.

3.1 Distance Functions

The distances defined in this section are presented in [7] by its authors. They adapted them from the time series distances that can be found in [8], by adding the y coordinate or the time t into the formulas. Those time series distances are the *Euclidean Distance* and the *Short Time Series Distance*, which give the name to the adapted versions for trajectories.

Adapted Euclidean Distance The distance between two trajectories partially overlapped over time should be less than the distance between two trajectories non overlapped. Thus, we need to include the time factor in the distance calculation.

Given two time series r and s as follows

$$r = \{p_1, p_2, \ldots, p_n\}$$
$$s = \{q_1, q_2, \ldots, q_n\}$$

with their elements p_i and q_i represented as

$$p = (p_x, p_y, p_t)$$
$$q = (q_x, q_y, q_t)$$

we can define the Adapted Euclidean Distance between r and s as:

$$d_{EU}(r, s) = \sqrt{\sum_{i=1}^{n} d_e(r_i, s_i)}$$

where

$$d_e(p, q) = (1 + (p_t - q_t)^2) \cdot ((p_x - q_x)^2 + (p_y - q_y)^2)$$

Adapted Short Time Series Distance In order to apply the original *short time series* distance to trajectories, it is enough to add the y coordinate to the distance

formula. Thus, the Adapted Short Time Series Distance is defined as:

$$d_{STS_A}(r,s) = \sqrt{\sum_{i=1}^{n} d_s(r_i, s_i)}$$

where

$$d_s(p,q) = \left[\left(\frac{q_{x_{i+1}} - q_{x_i}}{q_{t_{i+1}} - q_{t_i}} - \frac{p_{x_{i+1}} - p_{x_i}}{q_{t_{i+1}} - q_{t_i}}\right)^2\right] + \left[\left(\frac{q_{y_{i+1}} - q_{y_i}}{p_{t_{i+1}} - p_{t_i}} - \frac{p_{y_{i+1}} - p_{y_i}}{p_{t_{i+1}} - p_{t_i}}\right)^2\right]$$

3.2 Average Functions

Within the context of the MDAV microaggregation, the average function allows to compute the centroid of a cluster. The common statistical average function does not suit our needs, as we are working with trajectories and that would not provide realistic results. That means that if the centroids are computed this way, it may happen that the centroid point fall inside a building, which would be correct if the individual is a pedestrian, but not if it is an automobile. Thus, we assume that points can not be inside a building so, in order to avoid that, in [7] we propose two approaches based on the median function.

(X Median, Y Median) The first approach consists on calculating the median for the X coordinate of points of different trajectories that have the same timestamp t and fall into the same cluster. Then, the same operation is performed with the Y coordinate in order to obtain a trajectory that represents all the points in the given cluster.

For example, given the trajectories r, s and u defined as

$$r = (r_1, r_2, r_3)$$
$$s = (s_1, s_2, s_3)$$
$$u = (u_1, u_2, u_3)$$

where

$$r_1 = (3, 6, 1), r_2 = (1, 7, 2), r_3 = (8, 5, 3)$$
$$s_1 = (2, 9, 1), s_2 = (7, 8, 2), s_3 = (2, 4, 3)$$
$$u_1 = (0, 3, 1), u_2, = (6, 2, 2), u_3 = (9, 2, 3)$$

we calculate the median for the X and Y coordinate of the points with the same timestamp t, resulting in a single trajectory v. That is,

$$v = (v_1, v_2, v_3)$$

where

$$v_1 = (2, 6, 1), v_2 = (6, 7, 2), v_3 = (8, 4, 3)$$

(X Median, Y) The second approach is similar to the previous one. Again, it consists on calculating the median for the X coordinate of the same points of the different trajectories that fall into the same cluster. Then, instead of computing the median for the Y coordinate, search through all the trajectories in the same cluster for that X value and get the corresponding Y coordinate for that value. With this method, we obtain a trajectory with real points that represents all the trajectories of that cluster.

For example, given the trajectories r, s and u that fall into the same cluster, defined as

$$r = (r_1, r_2, r_3)$$
$$s = (s_1, s_2, s_3)$$
$$u = (u_1, u_2, u_3)$$

where

$$r_1 = (3, 6, 1), r_2 = (1, 7, 2), r_3 = (8, 5, 3)$$
$$s_1 = (2, 9, 1), s_2 = (7, 8, 2), s_3 = (2, 4, 3)$$
$$u_1 = (0, 3, 1), u_2, = (6, 2, 2), u_3 = (9, 2, 3)$$

we calculate the median for the X coordinate of the points with the same timestamp t and get the Y coordinate corresponding to that X value, resulting in a single trajectory v. That is,

$$v = (v_1, v_2, v_3)$$

where

$$v_1 = (2, 9, 1), v_2 = (6, 2, 2), v_3 = (8, 5, 3)$$

4 Evaluating Protected Trajectories

Evaluating a protection method is necessary and important, because it allows you to determine the protection level achieved and the amount of information that has been lost. In this section we detail the information loss and disclosure risk measures that we presented in [7].

4.1 Information Loss

As we stated before, the most common scenario is the one where the use that the protected data will have is unknown. Therefore, it is desired that the measures used to calculate the loss of information are as generic as possible in order to reflect the amount of information lost for a range of uses. For this reason, the information loss measure is defined in terms of three partial measures called IL_1, IL_2, and IL_3 that focus on different aspects, which are defined below.

***IL*$_1$** This partial measure is composed by the measures $IL_{1.1}$ and $IL_{1.2}$ defined below, and it is defined as:

$$IL_1 = \frac{IL_{1.1} + IL_{1.2}}{2}$$

- $IL_{1.1}$ is defined as the average of the difference between the means of both original and protected trajectories and is defined as follows:

$$IL_{1.1} = \frac{1}{2s}\left(\sum_{i=1}^{s} \frac{|\mu_{x_i} - \mu'_{x_i}|}{Max(|\mu_{x_i}|, |\mu'_{x_i}|)} + \frac{|\mu_{y_i} - \mu'_{y_i}|}{Max(|\mu_{y_i}|, |\mu'_{y_i}|)}\right)$$

- $IL_{1.2}$ is defined as the average of the difference between the autocorrelation function of both original and protected data. Formally,

$$IL_{1.2} = \frac{1}{4} \sum_{h=0,n/4,n/2,3n/4} \left(\frac{1}{s} \sum_{i=1}^{s} \frac{|\rho_i(h) - \rho_i(h)'|}{Max(|\rho_i(h)|, |\rho_i(h)'|)}\right)$$

  where s is the number of trajectories in the given data set, μ_x the average of the X coordinate, μ_y the average of the Y coordinate, and the $\rho_i(h)$ is the *Adapted Autocorrelation Function* defined as follows.
  Given the time series

$$r = \{p_1, p_2, \ldots, p_n\}$$

  with elements p_i with the following components

$$p = (p_x, p_y, p_t),$$

  we can define the Adapted Autocorrelation Function as:

$$\rho(h) = \frac{\gamma'_A(h)}{\gamma'_A(0)}$$

  where

$$\gamma'_A(h) = n^{-1} \cdot \sum_{i=1}^{n-|h|} \left(p_{x_{t+|h|}} - \mu_x\right) \cdot \left(p_{x_t} - \mu_x\right) + \left(p_{y_{t+|h|}} - \mu_y\right) \cdot \left(p_{y_t} - \mu_y\right)$$

and

$$\mu_x = \frac{1}{n} \sum_{i=1}^{n} p_x$$

$$\mu_y = \frac{1}{n} \sum_{i=1}^{n} p_y$$

***IL*$_2$** This other partial measure is defined as the absolute differences between the original and protected trajectories. That is,

$$IL_2 = \frac{1}{2 \times s \times n} \sum_{i=1}^{s \times n} \left(\frac{||x_i| - |x'_i||}{Max(|x_i|, |x'_i|)} + \frac{||y_i| - |y'_i||}{Max(|y_i|, |y'_i|)} \right)$$

where s is the number of trajectories in the given data set, n the number of points of a trajectory (i.e. its length), x and y the coordinates of a point that belongs to a non protected trajectory and x' and y' the coordinates of a point that belongs to a protected trajectory. Each pair of $x, y - x', y'$ has the same timestamp t.

***IL*$_3$** The last partial measure is called the *Simplified Space Distortion* measure that was defined in [7]. As the name suggests, it is a simplified version of the *Space Distortion* measure [9]. The SSD is defined as follows.

$$SSD = \sum_{i=1}^{s} \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}$$

where s is the number of trajectories of a given data set, x and y the coordinates of a point that belongs to a non protected trajectory and x' and y' the coordinates of a point that belongs to a protected trajectory. Each pair of $x, y - x', y'$ has the same timestamp t.

After the SSD calculation, a min-max range normalization is applied as it is desirable to have this value in the range [0, 1].

4.2 Disclosure Risk

In order to evaluate the protection level achieved by the MDAV microaggregation we use the disclosure risk measure. It provides a value that represents the risk of an attacker being able to establish links between individuals in the original and protected data sets. This process of establishing links between records of two files that correspond to the same individual is called *Record Linkage* [10, 11]. There are different approaches for the record linkage process, such as probabilistic record linkage

or the distance based record linkage [12]. In this article, we use the latter approach. The *Distance-Based Record Linkage* tries to establish those links by computing the distance between all the possible pairs between the original and protected data sets, by using a *one-against-all* approach, and taking the pair with the minimum distance between them as a correct match. Here, we use the two distance functions that are used in the microaggregation process. That is the *Adapted Euclidean Distance* and the *Adaptaed Short Time Series Distance*. Consider AEULD (Adapted Euclidean distance Linkage Disclosure) and ASTSLD (Adapted Short Time Series distance Linkage Disclosure), then, the disclosure risk (DR) is computed as

$$DR = Max(AEULD, ASTSLD)$$

The *Record Linkage* method for trajectories is shown in Algorithm 2.

Algorithm 2: Trajectory Record Linkage

Input: X: original data set, k: protected data set
Output: X': LP: linked pairs
1: **foreach** $a \in X$ **do**
2: $b' = arg\_min_{b \in X'} d_{tr}(a, b)$
3: $LP = LP \cup (a, b')$
4: **end foreach**

5 Overview of the Prototype

In this section we present the overview of the prototype built in order to apply the protection method described in Sect. 3.

As we already pointed out, the main goal of this work was to apply the protection method developed in our previous works [7, 13] to a real scenario. The scenario was reduced to a model of a little city with two robots being driven through the streets, instead of having real cars driving in real streets. This scenario was built in order to demonstrate the work done during the ARES (Advanced Research on Information Security and Privacy) project, which includes our work.

The tracking of the robots was performed by using a camera placed at the top of the structure, which was registering their moves. The tracking system allowed us to obtain the trajectories of the two robots, which are the ones used for this experiment. A sample of the obtained trajectories can be found in Table 1.

We developed a small GUI for the framework presented in [13] in which we can see the original trajectories for the two bots, as well as their protected equivalents. The GUI showing the original trajectories is presented in Fig. 1, and showing the protected ones in Fig. 2.

Table 1 Sample of a robot trajectory

Robot Id	Coord *X*	Coord *Y*	timestamp *t*
0	84	620	63452121868312
0	84	619	63452121868412
0	84	618	63452121868515
0	83	617	63452121868612
0	83	616	63452121868715
1	97	619	63452121860816
1	96	616	63452121860917
1	96	611	63452121861017
1	96	605	63452121861117
1	95	600	63452121861217

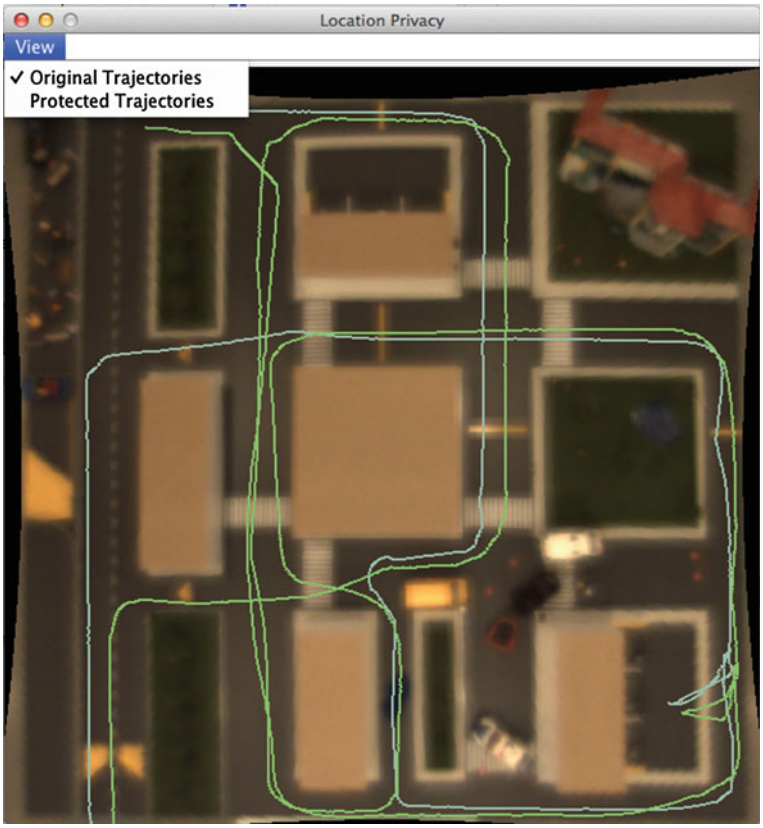


Fig. 1 GUI showing the original trajectories

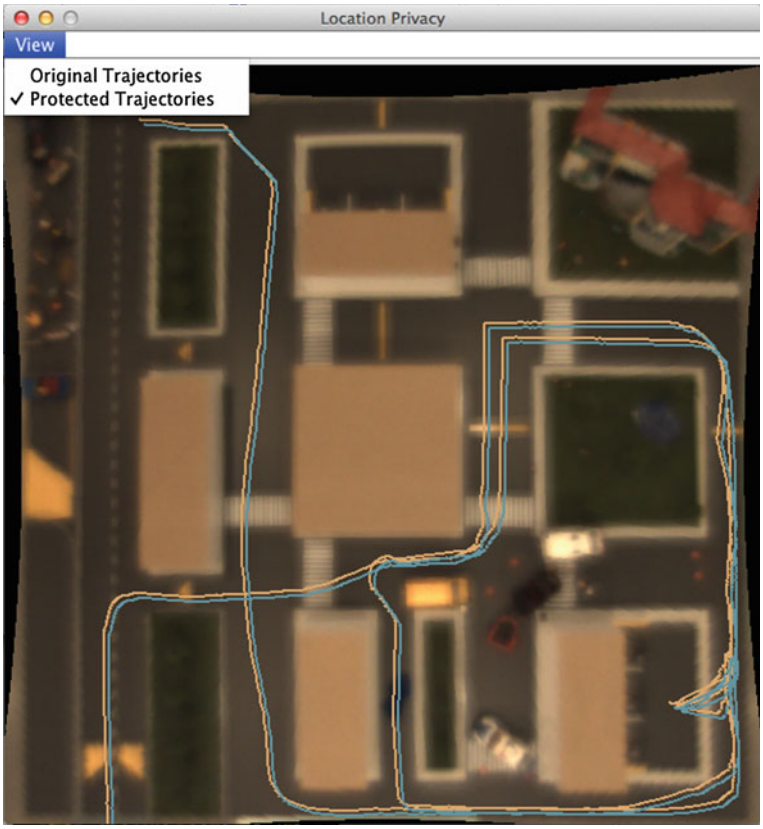


Fig. 2 GUI showing the protected trajectories

6 Conclusions

In this book chapter, we reviewed the protection method for trajectories based on the protection methods for time series [7]. We tested the protection method, as well as the evaluation framework [13], in a scenario close to real life, by using a scaled version of a city and two robots as moving entities. In addition, we developed a simple GUI in order to properly show the original and the protected trajectories, drawn on the aerial view of the city.

References

1. Ghinita, G.: Private queries and trajectory anonymization: a dual perspective on location privacy. Trans. Data Priv. **2**, 3–19 (2009), http://www.tdp.cat/issues/tdp.a018a09.pdf
2. Lin, D., Bertino, E., Cheng, R., Prabhakar, S.: Location privacy in moving-object environments. Trans. Data Priv. **2**, 21–46 (2009). http://www.tdp.cat/issues/tdp.a019a09.pdf

3. Nergiz, M.E., Atzori, M., Güç, B., Saygın, Y.: Towards trajectory anonymization: a generalization-based approach. Trans. Data Priv. **2**(1), 47–75 (2009)
4. Poolsappasit, N., Ray, I.: Towards achieving personalized privacy for location-based services. Trans. Data Priv. **2**(1), 77–99 (2009). http://dl.acm.org/citation.cfm?id=1556406.1556411
5. Damiani, M.L.: Third party geolocation services in lbs: Privacy requirements and research issues. Trans. Data Priv **4**(2), 55–72 (2011). http://dl.acm.org/citation.cfm?id=2019316.2019318
6. Ho, S.S., Ruan, S.: Preserving privacy for interesting location pattern mining from trajectory data. Trans. Data Priv. **6**, 185–198 (2013)
7. Martinez-Bea, S., Torra, V.: Trajectory anonymization from a time series perspective. In: Fuzzy Systems (FUZZ), 2011 IEEE International Conference on. pp. 401–408. IEEE (2011)
8. Nin, J., Torra, V.: Towards the evaluation of time series protection methods. Inf. Sci. **179**(11), 1663–1677 (2009)
9. Domingo-Ferrer, J., Sramka, M., Trujillo-Rasúa, R.: Privacy-preserving publication of trajectories using microaggregation. In: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS. pp. 26–33. SPRINGL '10, ACM, New York, NY, USA (2010), http://doi.acm.org/10.1145/1868470.1868478
10. Torra, V., Domingo-Ferrer, J.: Record linkage methods for multidatabase data mining. In: Torra, V. (ed.) Information Fusion in Data Mining, Studies in Fuzziness and Soft Computing, vol. 123, pp. 101–132. Springer, Berlin (2003)
11. Winkler, W.E.: Re-identification methods for masked microdata. In: Proceeding of the Privacy in Statistical Databases 2004, Springer LNCS 3050. pp. 216–230. Springer (2004)
12. Domingo-Ferrer, J., Torra, V.: Validating distance-based record linkage with probabilistic record linkage. In: Escrig, M., Toledo, F., Golobardes, E. (eds.) Topics in Artificial Intelligence. Lecture Notes in Computer Science, vol. 2504, pp. 207–215. Springer, Berlin (2002)
13. Martínez-Bea, S., Torra, V.: An evaluation framework for location privacy. CCIA **2**, 140–148 (2011)

Part V
Respondent Privacy: Social Networks

A Summary of k-Degree Anonymous Methods for Privacy-Preserving on Networks

Jordi Casas-Roma, Jordi Herrera-Joancomartí and Vicenç Torra

Abstract In recent years there has been a significant raise in the use of graph-formatted data. For instance, social and healthcare networks present relationships among users, revealing interesting and useful information for researches and other third-parties. Notice that when someone wants to publicly release this information it is necessary to preserve the privacy of users who appear in these networks. Therefore, it is essential to implement an anonymization process in the data in order to preserve users' privacy. Anonymization of graph-based data is a problem which has been widely studied last years and several anonymization methods have been developed. In this chapter we summarize some methods for privacy-preserving on networks, focusing on methods based on the k-anonymity model. We also compare the results of some k-degree anonymous methods on our experimental set up, by evaluating the data utility and the information loss on real networks.

1 Introduction

Nowadays, large amounts of data are being collected on social and other kinds of networks, which often contain personal and private information of users and individuals. Although basic processes are performed on data anonymization, such as removing names or other key identifiers, the remaining information can still be sensitive and useful for an attacker to re-identify users and individuals. To solve this

J. Casas-Roma (✉)
Universitat Oberta de Catalunya, Barcelona, Spain
e-mail: jcasasr@uoc.edu

J. Herrera-Joancomart
Universitat Autònoma de Barcelona, Bellaterra, Spain
e-mail: jherrera@deic.uab.cat

V. Torra
Institut d'Investigació en Intel·ligència Artificial,
Consejo Superior de Investigaciones Científicas Campus de la UAB,
08193 Bellaterra, Catalonia, Spain
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_13

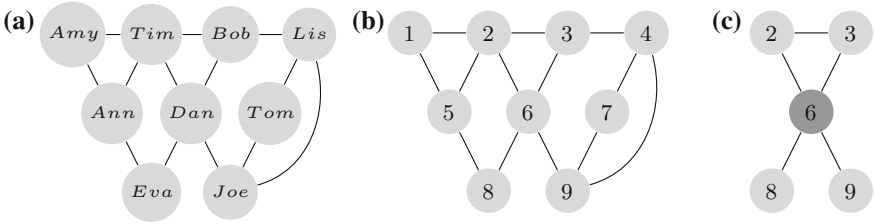


Fig. 1 Naïve anonymization of a toy network, where G is the original graph, $\tilde{G}$ is the naïve anonymous version and $\tilde{G}_{Dan}$ is Dan's 1-neighbourhood. **a** G, **b** $\tilde{G}$, **c** $\tilde{G}_{Dan}$

problem, methods which introduce noise to the original data have been developed in order to hinder the subsequent processes of re-identification. A natural strategy for protecting sensitive information is to replace identifying attributes with synthetic identifiers. We refer to this procedure as simple or naïve anonymization. This common practice attempts to protect sensitive information by breaking the association between the real-world identity and the sensitive data.

Figure 1a shows a toy example of a social network, where each vertex represents an individual and each edge indicates the friendship relation between them. Figure 1b presents the same graph after a naïve anonymization process, where vertex identifiers have been removed and the graph structure remains the same. One can think users' privacy is secure, but an attacker can break the privacy and re-identify a user on the anonymous graph. For instance, if an attacker knows that Dan has four friends and two of them are friends themselves, then he can construct the 1-neighbourhood of Dan, depicted in Fig. 1c. From this sub-graph, the attacker can uniquely re-identify user Dan on anonymous graph. Consequently, user's privacy has been broken by the attacker.

Two types of attacks have been proposed which show that identity disclosure would occur when it is possible to identify a sub-graph in the released graph in which all the vertex identities are known [2]. In the active attack an adversary creates k accounts and links them randomly, then he creates a particular pattern of links to a set of m other users that he is interested to monitor. The goal is to learn whether two of the monitored vertices have links between them. When the data is released, the adversary can efficiently identify the sub-graph of vertices corresponding to his k accounts with high probability. With as few a $k = \mathcal{O}(log(n))$ accounts, an adversary can recover the links between as many as $m = \mathcal{O}(log^2(n))$ vertices in an arbitrary graph of size n. The passive attack works in a similar manner. It assumes that the exact time point of the released data snapshot is known, and that there are k colluding users who have a record of what their links were at that time point. Other attacks on naively anonymized network data have been developed, which can re-identify vertices, disclose edges between vertices, or expose properties of vertices (e.g., vertex features). These attacks include: matching attacks, which use external knowledge of vertex features [26, 39, 41]; injection attacks, which alter the network prior to publication [2]; and auxiliary network attacks, which use publicly available

networks as an external information source [28]. To solve these problems, methods which introduce noise to the original data have been developed in order to hinder the subsequent processes of re-identification.

In this chapter we will summarize some methods for privacy-preserving on networks, specifically, we will focus on methods based on the concept of k-anonymity model. This model is widely used for data privacy, both for relational and graph-formatted data. We will also compare four k-anonymous methods in terms of data utility and information loss on undirected and unlabelled real networks.

This chapter is organized as follows. In Sect. 2, we review the state of the art of anonymization on networks, specifically the k-degree anonymous methods. Section 3 introduces the four tested algorithms for k-degree anonymity on networks. Then, in Sect. 4, we compare our tested algorithms among them, in terms of information loss and data utility, and discuss the results. Lastly, in Sect. 5, we present the conclusions.

1.1 Notation

Let $G = (V, E)$ be a simple, undirected and unlabelled graph, where V is the set of vertices and E the set of edges in G. We define $n = |V|$ to denote the number of vertices and $m = |E|$ to denote the number of edges. We use d to define the degree sequence of G, where d is a vector of length n and d_i is the value of i-th element, that is, the degree of vertex $v_i \in V$. We refer to the ordered degree sequence as a monotonic non-decreasing sequence of the vertex degrees, that is $d_i \leq d_j \ \forall i < j$. We denote the set of 1-neighbourhood of vertex v_i as $\Gamma(v_i)$, i.e., $\Gamma(v_i) = \{v_j : (v_i, v_j) \in E\}$. Finally, we designate $G = (V, E)$ and $\tilde{G} = (\tilde{V}, \tilde{E})$ to refer the original and the anonymous graphs, respectively.

2 Privacy-Preserving on Networks

Zhou and Pei [39] noticed that to define the problem of privacy preservation in publishing social network data, we need to formulate the following issues: Firstly, we need to identify the privacy information to be preserved. Secondly, we need to model the background knowledge that an adversary may use to attack the privacy. Thirdly, we need to specify the usage of the published social network data so that an anonymization method can try to retain the utility as much as possible while the privacy information is fully preserved.

Regarding to the privacy information to be preserved, we point out three main categories of privacy breaches in social networks:

1. *Identity disclosure* occurs when the identity of an individual who is associated with a vertex is revealed.

2. *Link disclosure* occurs when the sensitive relationship between two individuals is disclosed.
3. *Attribute disclosure* which seeks not necessarily to identify a vertex, but to reveal sensitive labels of the vertex. The sensitive data associated with each vertex is compromised.

Identity disclosure and link disclosure apply on all types of networks. However, attribute disclosure only applies on edge-labelled networks. In addition, link disclosure can be considered a special type of attribute disclosure, since edges can be seen as a vertex attributes. In this text, we will focus on identity disclosure.

From a high level view, there are three general families of methods for achieving network data privacy. The first family encompasses "graph modification" methods. These methods first transform the data by edges or vertices modifications (adding and/or deleting) and then release them. The data is thus made available for unconstrained analysis. The second family encompasses "generalization" or "clustering-based" approaches. These methods can be essentially regarded as grouping vertices and edges into partitions called super-vertices and super-edges. The details about individuals can be hidden properly, but the graph may be shrunk considerably after anonymization, which may not be desirable for analysing local structures. The generalized graph, which contains the link structures among partitions as well as the aggregate description of each partition, can still be used to study macro-properties of the original graph. Among others, [3, 5, 14, 20, 29] are interesting approaches to generalization concept. Finally, the third family encompasses "privacy-aware computation" methods, which do not release data, but only the output of an analysis computation. The released output is such that it is very difficult to infer from it any information about an individual input datum. For instance, differential privacy [16] is a well-known privacy-aware computation approach. Differential private methods refer to algorithms which guarantee that individuals are protected under the definition of differential privacy, which imposes a guarantee on the data release mechanism rather than on the data itself. The goal is to provide statistical information about the data while preserving the privacy of users. Interesting works can be found, among others, in [15, 21, 22].

2.1 Graph Modification Approaches

Graph modification approaches anonymize a graph by modifying (adding and/or deleting) edges or vertices in a graph. These modifications can be made randomly or in order to fulfil some desired constraints. The first methods are called randomization methods and are based on adding random noise in the original data. They have been well investigated for relational data. Naturally, edge randomization can also be considered as an additive-noise perturbation. Notice that the randomization approaches protect against re-identification in a probabilistic manner. Hay et al. [19] proposed a method to anonymize unlabelled graphs based on randomly removing m

edges and then randomly adding m fake edges. Ying and Wu [36] propounded two algorithms specifically designed to preserve spectral characteristics of the original graph. Ying et al. [35] presented a method which divides the graph into blocks according to the degree sequence and implements modifications (by adding and removing edges) on the vertices at high risk of re-identification, not at random over the entire set of vertices. Boldi et al. [4] introduced a new anonymization approach that is based on injecting uncertainty in social graphs (they add or remove edges partially with a certain probability) and publishing the resulting uncertain graphs. Other approaches consider the degree sequence of the vertices or other structural graph characteristics (for example, transitivity or average distance between pairs of vertices) as important features which the anonymization process has to keep as equal as possible on anonymized network [17, 37].

2.2 k-Anonymity Model

Other ways to anonymize consider graph modification methods to meet desired privacy constraints. The notion of k-anonymity [30, 32] is included in this group, though it was introduced for the privacy preservation on relational data. Formally, the k-anonymity model is defined as follows. Let $RT(A_1, \ldots, A_n)$ be a table and QI_{RT} be the quasi-identifier associated with it. RT is said to satisfy k-anonymity if and only if each sequence of values in $RT[QI_{RT}]$ appears with at least k occurrences in $RT[QI_{RT}]$. The k-anonymity model indicates that an attacker cannot distinguish between different k records although he manages to find a group of quasi-identifiers. Therefore, the attacker cannot re-identify an individual with a probability greater than $\frac{1}{k}$. In general, the higher the k value, the greater the anonymization and also the information loss. Ying et al. [35] demonstrated that deliberate k-anonymization can preserve structural properties of networks much better than the randomization techniques.

The k-anonymity model can be applied using different quasi-identifiers when dealing with networks rather than relational data. A widely used option is to consider the vertex degree as a quasi-identifier, i.e., this model presumes that the only possible attack is when the attacker knows the degree of some target vertices. This corresponds to k-degree anonymity. Therefore, if some vertices are re-identified using this information, then we have an information leakage. Liu and Terzi [26] developed a method to create a k-degree anonymous network $\tilde{G} = (V, \tilde{E})$ from the original network $G = (V, E)$ and an integer k, where $\tilde{E} \cap E \approx E$. Their method is based on anonymizing the degree sequence by linear programming techniques. Casas-Roma et al. [8] presented a method based on evolutionary algorithms, which anonymizes the degree sequence and then translates the modifications to the edge set. Chester et al. [10, 12] also considered the k-degree anonymity problem, but they modified the network structure by adding new edges between fake and real vertices or between fakes vertices. Under the constraint of minimum vertex additions, they show that on vertex-labelled networks, the problem is NP-complete. Casas-

Roma et al. [6] introduced an algorithm specifically designed for k-degree anonymity on large networks. The authors construct a k-degree anonymous network by the minimum number of edge modifications using univariate micro-aggregation to anonymize the degree sequence, and then they modify the graph structure using basic operations for graph modification to meet the k-degree anonymous sequence.

Chester et al. [11] introduced the concept of k-subset-degree anonymity as a generalization of the notion of k-degree-anonimity. In k-subset-anonymity problem the goal is to anonymize a given subset of vertices, while adding the fewest possible number of edges. Formally, k-degree-subset-anonymity problem is defined as given an input graph $G = (V, E)$ and an anonymous subset $X \subseteq V$, produces an output graph $\tilde{G} = (V, E \cup \tilde{E})$ such that X is k-degree-anonymous and $|\tilde{E}|$ is minimized. They presented an algorithm to k-subset-degree-anonymity which is based on using the degree constrained sub-graph satisfaction problem. For unlabelled networks, they give a near-linear algorithm ($\mathcal{O}(nk)$). The output of the algorithm is an anonymized version of G where enough edges have been added to ensure all the vertices in X have the same degree as at least $k - 1$ others.

Zhou and Pei [39, 40] introduced the 1-neighbourhood sub-graph of the objective vertices as a quasi-identifier. For a vertex $u \in V$, u is k-anonymous in G if there are at least $k - 1$ other vertices $v_1, \ldots, v_{k-1} \in V$ such that $\Gamma(u), \Gamma(v_1), \ldots, \Gamma(v_{k-1})$ are isomorphic. G is k-anonymous if every vertex is k-anonymous in G. It is called k-neighbourhood anonymity. Tripathy and Panda [33] noted that their algorithm cannot handle the situations in which an adversary has knowledge about vertices in the second or higher hops of a vertex, in addition to its immediate neighbours. To handle this problem, they proposed a modification to their algorithm to handle such situations. In addition, the time complexity of their algorithm is less than that of Zhou and Pei. Zou et al. [41] considered all structural information about a target vertex and propounded a new model called k-automorphism. Hay et al. [20] go a step further and proposed a method, named k-candidate anonymity, that uses queries as quasi-identifier. In this method, a vertex v_i is k-candidate anonymous to question Q if there are at least $k - 1$ others vertices in the network with the same answer. Cheng et al. [9], in their work on k-isomorphism, formed k pairwise isomorphic sub-graphs to achieve protection against two specific classes of attacks. Wu et al. [34] introduced the k-symmetry model, wherein for any vertex v, there exists at least $k-1$ other vertices to which v can be mapped using an automorphism of the underlying graph. Kapron et al. [23] analysed the problem of anonymizing an edge-labelled network. They considered the label sequence $S_v = (\ell_1, \ell_2, \ldots, \ell_m)$ of a vertex v as some ordering of the labels of the edges incident on v. Lastly, Stokes and Torra [31] introduced the concept of n-confusion as a generalization of k-anonymity and a new definition of (k,ℓ)-anonymous graph, which they proved to have severe weaknesses. The authors also presented a set of algorithms for k-anonymization of graphs.

When there is little diversity in the sensitive attributes inside an equivalence class, it is possible to obtain information from anonymized data. Although there are k indistinguishable records in each equivalence class, if the information in sensitive attributes is the same, then it is possible to infer information unless the attacker does not know exactly which record it is. The ℓ-diversity model [27] alleviates the

problem of sensitive attribute disclosure. It ensures that the sensitive attribute value in each equivalence class are diverse. But an attacker can also infer some sensible information from similarity or skewness attack [25]. This leads to t-closeness [25], which is another privacy definition that considers the sensitive attribute distribution in each class. There are other privacy definitions of this flavour, but they are all been criticized for being ad hoc [38].

Chester et al. [13] study the complexity of anonymization on different kinds of network (labelled, unlabelled and bipartite). For general, edge-labelled graphs, label sequence subset anonymization (and thus table graph anonymization, k-neighbourhood anonymity, i-hop anonymity and k-symmetry) are NP-complete for $k \geq 3$. For bipartite, edge-labelled graphs, label sequence subset anonymization is in P for $k = 2$ and is NP-complete for $k \geq 3$. For bipartite, unlabelled graphs, degree-based subset anonymization is in P for all values of k. And for general, vertex-labelled graphs, they show that vertex label sequence-based anonymization, and consequently t-closeness, is NP-complete.

3 k-Degree Anonymous Methods

We have selected four relevant methods for k-degree anonymity on networks. In subsequent sections, we will analyse these methods and compare the empirical results on real networks. Firstly, Liu and Terzi defined the concept of k-degree anonymity and presented their method in [26]. Secondly, Casas-Roma et al. introduced two algorithms, the first one based on evolutionary algorithms [8] and the second one based on univariate micro-aggregation [6]. Lastly, Chester et al. propounded an algorithm based on vertex and edge addition [12]. All methods achieve the same privacy level, since they presuppose the same adversary knowledge and apply the same concept to preserve the network's privacy. Therefore, the evaluation of the results is interesting to compare the data utility and information loss on anonymous datasets.

3.1 Preliminaries

The degree sequence is an interesting tool since the concept of k-degree anonymity for a network can be directly mapped to its degree sequence, as Liu and Terzi showed in [26] and we recall in the following definitions:

Definition 1 A vector of integers V is k-anonymous if every distinct value $v_i \in V$ appears at least k times.

Definition 2 A network $G = (V, E)$ is k-degree anonymous if the degree sequence of G is k-anonymous.

Accordingly to Definition 2, the degree sequence is a key point when dealing with k-degree anonymity on networks. Regarding to the degree sequence, notice that:

- The number of elements is n, which represents the number of vertices.
- Each $d_i \in d$ must be an integer in the range $[0, n-1]$, since each d_i is the degree of vertex v_i.
- $\sum_{i=1}^{n} d_i = 2m$, since each edge is counted twice in the degree sequence. Therefore, $\sum_{i=1}^{n} d_i = \sum_{i=1}^{n} \tilde{d}_i$ if we want to keep the same number of edges in the anonymous graph.

The construction of the k-anonymous degree sequence determines the privacy level. Moreover, the distance between the original and the anonymous degree sequences is critical in terms of data utility and information loss. An optimal sequence has to provide the requested k-anonymity level and also has to minimize the distance from the original degree sequence. Some of our tested methods use Eq. 1 to compute the distance between the original degree sequence and the anonymous one.

$$\Delta = \sum_{i=1}^{n} |\tilde{d}_i - d_i| \tag{1}$$

3.2 A Dynamic Programming Algorithm

Liu and Terzi [26] developed a method based on adding and removing edges from the original graph $G = (V, E)$ in order to construct a new graph $\tilde{G} = (\tilde{V}, \tilde{E})$, which fulfil k-degree anonymity model and the vertex set remains the same, i.e., $V = \tilde{V}$. Their approach is two-step based: in the first step the degree anonymization problem is considered, and in the second step the graph construction problem is dealt.

Degree anonymization. Given the degree sequence d of the original input graph $G = (V, E)$, the algorithm outputs a k-anonymous degree sequence ($\tilde{d}$) such that the degree-anonymization cost Δ computed by Eq. 1 is minimized. The authors proposed three approximation techniques to solve the degree anonymization problem. They first gave a dynamic-programming algorithm (DP) that solves the degree anonymization problem optimally in time $\mathcal{O}(n^2)$. Then, they showed how to modify it to achieve linear-time complexity. Finally, they also gave a greedy algorithm that runs in time $\mathcal{O}(nk)$.

Graph construction. The authors presented two approaches to resolve the graph construction problem. The first approach considers the following problem: Given the original graph $G = (V, E)$ and the desired k-anonymous degree sequence $\tilde{d}$ (output by the the previous step), they construct a k-degree anonymous graph $\tilde{G} = (V, \tilde{E})$ with $\tilde{E} \cap E = E$ and degree sequence equal to $\tilde{d}$. Notice that the problem definition implies that only edge addition operations are allowed. The algorithm for solving this problem was called *SuperGraph*. It takes as inputs the original graph G and the desired k-degree anonymous sequence $\tilde{d}$, operates on the sequence of additional degrees $\tilde{d} - d$ and outputs a super-graph of the original graph, if such graph exists.

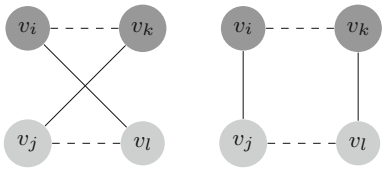


Fig. 2 Valid swap operation among vertices v_i, v_j, v_k and v_l. *Dashed lines* represent deleted edges while *solid lines* are the added ones

The requirement that $\tilde{E} \cap E = E$ may be too strict to satisfy. Thus, the second approach considers a relaxed requirement where $\tilde{E} \cap E \approx E$, which means that most of the edges of the original graph appear in the degree-anonymous graph as well, but not necessarily all of them. The authors called this version of the problem the "Relaxed Graph Construction" problem. The *ConstructGraph* algorithm with input $\tilde{d}$, outputs a simple graph $\tilde{G}_0 = (V, \tilde{E}_0)$ with degree sequence exactly $\tilde{d}$, if such graph exists. Although $\tilde{G}_0$ is k-degree anonymous, its structure may be quite different from the original graph $G = (V, E)$. The *GreedySwap* algorithm inputs $\tilde{G}_0$ and G, and transforms $\tilde{G}_0$ into $\tilde{G} = (V, \tilde{E})$ with degree sequence equal to $\tilde{d}$ and $\tilde{E} \cap E \approx E$ using greedy heuristic techniques. At each step i, the graph $\tilde{G}_{i-1} = (V, \tilde{E}_{i-1})$ is transformed into $\tilde{G}_i = (V, \tilde{E}_i)$ such that the degree sequences are equal and $|\tilde{E}_i \cap E| > |\tilde{E}_{i-1} \cap E|$. The transformation is made using *valid swap* operations, which are defined by four vertices v_i, v_j, v_k and v_l of $\tilde{G}_i = (V, \tilde{E}_i)$ such that (v_i, v_k) and $(v_j, v_l) \in \tilde{E}_i$, and (v_i, v_j) and $(v_k, v_l) \notin \tilde{E}_i$ or (v_i, v_l) and $(v_j, v_k) \notin \tilde{E}_i$. A valid swap operation transforms $\tilde{G}_i$ to $\tilde{G}_{i+1}$ by updating the edges $\tilde{E}_{i+1} \leftarrow \tilde{E}_i \setminus \{(v_i, v_k), (v_j, v_l)\} \cup \{(v_i, v_j), (v_k, v_l)\}$ or $\tilde{E}_{i+1} \leftarrow \tilde{E}_i \setminus \{(v_i, v_k), (v_j, v_l)\} \cup \{(v_i, v_l), (v_j, v_k)\}$, as we depict in Fig. 2.

3.3 An Univariate Micro-aggregation Approach

Univariate Micro-aggregation for Graph Anonymization[1] (in short, UMGA) algorithm was proposed in [6] and it was designed to achieve k-degree anonymity on large networks. The algorithm performs modifications to the original network only on edge set (E). Hence, the vertex set (V) does not change during anonymization process. In a similar way to the previous method, it is based on a two-step approach.

Degree sequence anonymization. It constructs a k-degree anonymous sequence $\tilde{d} = \{\tilde{d}_1, \ldots, \tilde{d}_n\}$ from the degree sequence $d = \{d_1, \ldots, d_n\}$ of the original network $G = (V, E)$ using Definition 1. This method uses the optimal univariate micro-aggregation [18] to achieve the best group distribution and then it computes the value for each group that minimizes the distance Δ computed by Eq. 1 from the original degree sequence.

Without loss of generality, the authors assume d to be an ordered degree sequence of the original network. Otherwise, they apply a permutation f to the sequence

[1] Source code available at: http://deic.uab.cat/~jcasas/.

to reorder the elements. Let k be an integer such that $1 \leq k < n$ which is the k-degree anonymity value. In order to apply the optimal univariate micro-aggregation, and according to Hansen and Mukherjee [18], the authors construct a new directed network $H_{k,n}$ and get the optimal partition which is exactly the set of groups that corresponds to the arcs of the shortest path from vertex 0 to vertex n on this network. They denote by g the optimal partition, where g has $\frac{n}{k} \leq p \leq \frac{n}{2k-1}$ groups and each of them (g_j) has between k and $2k-1$ items. Obviously, each $d_i \in d$ belongs to a specific group g_j.

Next, the algorithm computes the specific value for each group g_j, since the mean value of all group members $d_i \in g_j$ can be a real number and an integer number is needed in the degree sequence. Using the floor or ceiling functions to round these values, the total number of edges in each group g_j (computed as the sum of all $d_i \in g_j$) can be the same, higher (which means some new edges are needed) or smaller (which means some edges have to be deleted). To optimally resolve this operation two methods are proposed to achieve the best combination on reasonable time: firstly, the exhaustive method explores all possible combinations until it finds an optimal solution. Secondly, the greedy method uses a probability distribution to find a quasi-optimal (in many cases, the optimal) solution in a faster way.

Graph modification. It builds a new network $\tilde{G} = (V, \tilde{E})$ where its degree sequence is equal to $\tilde{d}$ by using basic edge modification operations. These operations allow it to modify the network's structure according to the anonymized degree sequence $(\tilde{d})$. By Definition 2 the anonymized network $\tilde{G}$ will be k-degree anonymous.

In order to modify the edge set of a given network, the authors define three basic operations: edge switch, edge removal and edge addition. The *edge switch* between three vertices can be defined as follows: if $v_i, v_j, v_k \in V$, $(v_i, v_k) \in E$ and $(v_j, v_k) \notin E$, we can delete (v_i, v_k) and create (v_j, v_k), as shown in Fig. 3a. The *edge removal* is defined as follows: we select four vertices $v_i, v_j, v_k, v_l \in V$ where $(v_i, v_k) \in E$, $(v_j, v_l) \in E$ and $(v_k, v_l) \notin E$. We delete edges (v_i, v_k) and (v_j, v_l), and create a new edge (v_k, v_l), as depicted on Fig. 3b. Finally, the *edge addition* is defined as follows: we select two vertices $v_i, v_j \in V$ where $(v_i, v_j) \notin E$ and create it. It is presented in Fig. 3c.

The selection of the auxiliary edges is an important feature, since adding or removing important edges is critical for network structure and information flow. For instance, adding or removing a bridge-like edge may considerably reduce or increase

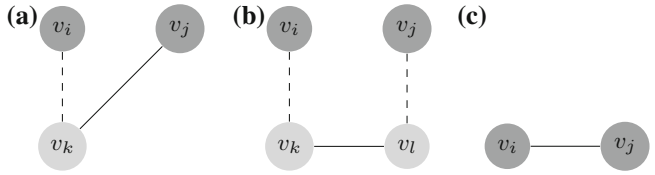


Fig. 3 Basic operations for network modification with vertex invariability. *Dashed lines* represent deleted edges while *solid lines* are the added ones. **a** Edge switch, **b** Edge removal, **c** Edge addition

the average distance and the shortest paths of the entire network. Two approaches were presented to select the auxiliary edges needed for graph modification process: the first one is based on random edge selection, which is the fastest way to select the auxiliary edges. The second approach is based on selecting auxiliary edges by considering the relevance of each edge according to edge neighbourhood centrality (NC) [7], which identifies the most important edges on a network with low complexity ($\mathcal{O}(m)$). Obviously, this approach leads the process to a low information loss results.

3.4 Vertex Addition Method

Chester et al. [10, 12] focused on creating a k-degree-anonymous graph $\tilde{G} = (V \cup \tilde{V}, E \cup \tilde{E})$ from the original one $G = (V, E)$. In $\tilde{G}$, the authors require that all the original vertices (V) are k-degree-anonymous. They also require that the new vertices are concealed as well so that they cannot be readily identified and removed from the graph in order to recover G, i.e., $V \cup \tilde{V}$ is k-degree-anonymous in $\tilde{G}$. They seek to minimise $|\tilde{V}|$, while maintaining the constraint that $E \subseteq \tilde{V} \times (V \cup \tilde{V})$.

Their method introduces fake vertices into the network and links them to each other and to real vertices in order to achieve the desired k-anonymity value. The authors introduced an $\mathcal{O}(kn)$ k-degree anonymization algorithm for unlabelled graphs based on dynamic programming and prove that, on any arbitrary graph, the minimisation of $|\tilde{V}|$ is optimal within an additive factor of k. For a special class of graphs that is likely to include social networks, the algorithm is optimal within 1 for reasonable values of k.

At a high level, the algorithm proceeds in three stages. First, Chester et al. designed a recursion to group the vertices of V by target degree (the degree they will have in $\tilde{G}$). The recursion establishes a grouping such that the *max deficiency*, a parameter in determining with how many vertices V must be augmented, is minimised. A dynamic programming with cost $\mathcal{O}(nk)$ is used to evaluate the recursion. The second stage is to determine precisely how many vertices with which we wish to augment V in order to guarantee that they can k-anonymize all of $\tilde{G}$. This number is a function of k and *max deficiency*. Finally, the algorithm introduces a particular means of adding new edges, each of which has at least one endpoint in $\tilde{G}$, with the objective of satisfying all the target degrees established during the recursion stage and k-anonymizing the new vertices added during the second stage. A critical property of this approach is that the edges are added in such a manner as to guarantee tractability of the problem of k-anonymizing the new vertices, a problem which may be hard in the general case.

3.5 An Evolutionary Algorithm Approach

Evolutionary Algorithm for Graph Anonymization[2] (in short, EAGA) [8] is a method focused on constructing a k-degree anonymous graph using evolutionary algorithms. A high-level description of this proposal allows us to structure it in two steps, in a similar way to the previous approaches.

Obtaining the k-degree anonymous sequence. In the first step, from the original degree sequence $d = \{d_1, \ldots, d_n\}$ of $G = (V, E)$, it constructs a new sequence $\tilde{d}$ which is k-degree anonymous and tries to minimize the distance Δ from the original sequence computed by Eq. 1.

As we have commented, the anonymization of the degree sequence is computed by an evolutionary algorithm. The population is initialised from original degree sequence and many iterations are performed until a valid solution is found. The mutation process, which is responsible of the new candidates generation, applies a basic edge switch at each step (i.e., it adds one to an element of the sequence and subtracts one to another element of the sequence). This basic operation represents a change on a vertex of an edge, which is the most basic edge modification on a graph. For example, if an edge (v_i, v_k) is modified by replacing one vertex, one can obtain (v_j, v_k). This edge modification is represented on the degree sequence as a subtraction on vertex v_i (because it decreases its degree) and a addition on vertex v_j (because it increases its degree). It is important to note that this algorithm does not use crossover since this operation systematically breach the rule that preserves the number of edges of the graph, generating invalid candidates. The authors state the performance of the algorithm would be affected by the inclusion of this type of evolution and improvements would not occur in time or quality of the solution found. When candidate generation is done, the algorithm evaluates the candidates in order to find the best one. The score of each candidate is determined by the fitness function, which is a two-state function: if the k value of the candidate is lower than the desired one, the fitness function considers the dispersion in the degree histogram and the number of vertices which belong to groups between 0 and $k - 1$ in the degree histogram, i.e., the number of vertices which does not fulfil de k-degree anonymity. This step is called "expansion" since the candidates tend to expand on the representation space trying to find a valid solution. On the contrary, if the k value of the candidate is equal or greater than the desired one, the fitness function only considers the distance from the original degree sequence. This step is called "retraction", since the candidates tend to move close to the original degree sequence. The candidate selection uses the steady-state model to choose the individuals which will survive to the next generation.

Modifying the original graph. In the second step, the algorithm constructs a graph $\tilde{G} = (\tilde{V}, \tilde{E})$ where $\tilde{V} = V$, $\tilde{E} \cap E \approx E$ and the degree sequence is equal to $\tilde{d}$. The difference between the anonymized and the original degree sequences $\tilde{d} - d$ points to vertices which have to increase or decrease their degree. Thus, some edges

[2] Refer footnote 1.

have to be added or removed from/to these vertices. The algorithm applies these modifications by edge switch, which consists on removing an edge $(v_i, v_k) \in E$, where v_i belongs to vertices which have to decrease their degree, and adding a new edge (v_j, v_k), where v_j belongs to vertices which have to increase their degree, as we show in Fig. 3a.

4 Experimental Results

In this section we will compare the result of anonymizing processes using the four k-degree anonymous methods presented in Sect. 3. We apply all algorithms on the same data with the same parameters and compare the results in terms of information loss and data utility. We use several structural and spectral measures in order to quantify the information loss from distinct perspectives or network's characteristics. It is important to note that the privacy level is the same for all algorithms, as we compare results with the same k value. UMGA algorithm is applied using the neighbourhood centrality edge selection.

4.1 Tested Networks

Table 1 shows a summary of the networks' main features, including number of vertices, number of edges, average degree and default k-anonymity value. US politics book data (polbooks) [24] is a network of books about US politics published around the 2004 presidential election and sold by the on-line bookseller Amazon. Edges between books represent frequent co-purchasing of books by the same buyers. Political blogosphere data (polblogs) [1] compiles the data on the links among US political blogs. Both of them are undirected and unlabelled networks.

4.2 Measures

In order to compare the algorithms, we use several well-known structural and spectral measures [4, 12, 35, 36]. The first structural measure is *harmonic mean of the shortest distance* (h). It is an evaluation of connectivity, similar to the average distance or

Table 1 General properties of tested networks: number of vertices ($|V|$), number of edges ($|E|$), average degree ($\langle deg \rangle$) and default k-anonymity value (k)

Network	$\lvert V\rvert$	$\lvert E\rvert$	$\langle deg \rangle$	k
Polbooks	105	441	8.40	1
Polblogs	1,222	16,714	27.31	1

average path length. The inverse of the harmonic mean of the shortest distance is also known as the global efficiency, and it is computed by Eq. 2, where $d(v_i, v_j)$ is the length of the shortest path from v_i to v_j, meaning the number of edges along the path.

$$\frac{1}{h} = \frac{1}{n(n-1)} \sum_{\substack{i,\, j\, =\, 1 \\ i \neq j}}^{n} \frac{1}{d(v_i, v_j)} \tag{2}$$

Modularity (Q) indicates the goodness of the community structure. It is defined as the fraction of all edges that lie within communities minus the expected value of the same quantity in a network in which the vertices have the same degree, but edges are placed at random without regard for the communities.

Transitivity (T) is one type of clustering coefficient, which measures and characterizes the presence of local loops near a vertex. It measures the percentage of paths of length 2 which are also triangles.

Lastly, *sub-graph centrality* (SC) is used to quantify the centrality of vertex v_i based on the sub-graphs. Formally:

$$SC = \frac{1}{n} \sum_{i=1}^{n} SC_i = \frac{1}{n} \sum_{i=1}^{n} \sum_{k=0}^{\infty} \frac{P_i^k}{k!} \tag{3}$$

where P_i^k is the number of paths from v_i to v_i with length k.

Moreover, two spectral measures which are closely related to many network characteristics [36] are used. *The largest eigenvalue of the adjacency matrix A* (λ_1) where λ_i are the eigenvalues of A and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. The eigenvalues of A encode information about the cycles of a network as well as its diameter. *The second smallest eigenvalue of the Laplacian matrix L* (μ_2) where μ_i are the eigenvalues of L and $0 = \mu_1 \leq \mu_2 \leq \cdots \leq \mu_m \leq m$. The eigenvalues of L encode information about the tree structure of G. μ_2 is an important eigenvalue of the Laplacian matrix and can be used to show how good the communities separate, with smaller values corresponding to better community structures.

4.3 Empirical Results

Results are disclosed in Table 2. Each row indicates the scored value for the corresponding measure and algorithm, and each column corresponds to an experiment with a different k-anonymity value. Each characteristic is reported from two to four times, corresponding to EAGA, UMGA, Liu and Terzi (indicated by L&T) and Chester et al. (indicated by Chester) algorithms. A bold row indicates the best algorithm for each measure and network. Values of Liu and Terzi algorithm are taken from Ying et al. [35] and values of Chester et al. algorithm are taken from [12]. Unfortunately, val-

ues for all measures and algorithms are not available. Perfect performance in a row would be indicated by achieving exactly the same score as in the original network (the $k = 1$ column). Although deviation is undesirable, it is inevitable due to the edge or vertex modification process.

The first tested network, Polbooks, is a small collaboration network. We present values for EAGA, UMGA and Liu and Terzi algorithms. As shown in Table 2, UMGA algorithm introduces less noise on all measures. It outperforms on all measures, producing half of the average error in some measures, for example, λ_1 or SC. EAGA algorithm achieves the second best results on λ_1, μ_2, h and Q, while Liu and Terzi algorithm carry out on T and SC.

Polblog is the second tested network, which is considerably larger than the first one. Values for Chester et al. algorithm are presented for h, T and SC (other values are not available from Chester et al. [12]). Like in the previous test, UMGA algorithm gets the best values on all measures, except on μ_2 where Liu and Terzi algorithm achieves the same value. For instance, the average error is 0.006 for UMGA on h, while it rises to 0.026 for Liu and Terzi algorithm, 0.039 for Chester et al. approach, and 0.071 for EAGA. Similar results appear on λ_1, T and SC. Liu and Terzi algorithm obtains the second best results on λ_1, h and T, while EAGA does on Q and SC. Chester et al. approach by vertex addition gets values close to others algorithms, though the predictable level of information loss is slightly larger than the ones obtained by UMGA and Liu and Terzi algorithms. Despite the fact that EAGA gets good results on some metrics, the average error outbursts in many others. For example, results on λ_1 and μ_2 are larger than others, pointing out a considerable spectral noise introduced by the anonymization process.

We note two important factors which can be decisive for the quality of the anonymous data: The first one is the number of modifications in the edge and vertex set. Clearly, it is important to minimise these values since keeping them close to the original ones will preserve the structural and spectral metrics. The second factor we point out is related to edge relevance. Some edges play an important role inside the network, and preserving them we will lead the process to a better data utility and lower information loss. For instance, a bridge-like edge is critical for the structure of the network and the information flow. Thus, preserving it will conduct the anonymization process to a low information loss results. Notice that UMGA is the only algorithm which considers the edge relevance.

5 Conclusions

We have reviewed recent studies on anonymization techniques for privacy-preserving publishing of graph-formatted data. The research and development of privacy-preserving social network analysis is still in its early stage compared with much better studied privacy-preserving data analysis for tabular data. In this chapter we have focused on methods related to k-anonymity model, specifically to k-degree anonymity methods. These methods consider the vertices degree as adversary's

Table 2 Results for EAGA, UMGA, Liu and Terzi (L&T) and Chester et al. (Chester) algorithms

Polbooks		$k=1$	2	3	4	5	6	7	8	9	10	$\langle\mathcal{E}\rangle$
λ_1	EAGA	11.93	12.04	12.01	12.04	11.95	12.05	12.01	11.72	10.84	11.45	0.230
	UMGA		**12.09**	**11.97**	**11.85**	**11.85**	**11.95**	**12.09**	**12.08**	**12.08**	**11.86**	**0.090**
	L&T		12.00	12.05	12.11	12.22	12.30	12.31	12.64	12.72	12.85	0.383
μ_2	EAGA	0.324	0.372	0.477	0.496	0.516	0.515	0.600	0.595	0.578	0.321	0.156
	UMGA		**0.360**	**0.451**	**0.453**	**0.453**	**0.383**	**0.599**	**0.524**	**0.524**	**0.640**	**0.147**
	L&T		0.430	0.450	0.600	0.600	0.790	0.630	0.650	0.970	0.880	0.312
h	EAGA	2.450	2.378	2.324	2.346	2.297	2.314	2.294	2.282	2.308	2.421	0.109
	UMGA		**2.416**	**2.371**	**2.379**	**2.379**	**2.418**	**2.312**	**2.350**	**2.350**	**2.312**	**0.077**
	L&T		2.350	2.320	2.280	2.280	2.230	2.270	2.260	2.200	2.190	0.167
Q	EAGA	0.402	0.399	0.387	0.387	0.383	0.387	0.379	0.379	0.387	0.389	0.014
	UMGA		**0.400**	**0.393**	**0.396**	**0.396**	**0.401**	**0.386**	**0.386**	**0.386**	**0.385**	**0.009**
	L&T		0.390	0.390	0.380	0.380	0.360	0.370	0.370	0.340	0.350	0.027
T	EAGA	0.348	0.343	0.330	0.324	0.281	0.300	0.288	0.283	0.245	0.299	0.044
	UMGA		**0.350**	**0.342**	**0.339**	**0.339**	**0.347**	**0.326**	**0.322**	**0.322**	**0.324**	**0.013**
	L&T		0.330	0.330	0.320	0.330	0.300	0.310	0.320	0.290	0.300	0.023
$SC(\times 10^3)$	EAGA	2.524	2.624	2.333	2.293	1.751	2.001	1.967	1.415	0.653	1.534	0.634
	UMGA		**2.774**	**2.358**	**2.224**	**2.224**	**2.338**	**2.363**	**2.389**	**2.389**	**2.110**	**0.204**
	L&T		2.480	2.560	2.530	2.760	2.440	2.680	3.600	3.580	4.120	0.431

(continued)

Table 2 continued

Polbooks		$k=1$	2	3	4	5	6	7	8	9	10	$\langle\mathcal{E}\rangle$
λ_1	EAGA	74.08	73.13	70.26	55.61	53.09	49.33	46.89	44.44	42.88	44.08	18.703
	UMGA		**73.93**	**73.81**	**73.92**	**73.95**	**73.74**	**73.80**	**73.75**	**73.63**	**73.61**	**0.256**
	L&T		74.89	74.50	75.16	75.10	76.32	75.82	76.67	77.42	78.42	1.758
μ_2	EAGA	0.168	0.168	0.168	0.692	0.674	0.748	0.754	0.690	0.858	0.757	0.517
	UMGA		0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.000
	L&T		0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.168	0.000
h	EAGA	2.506	2.677	2.623	2.596	2.592	2.588	2.595	2.565	2.572	2.575	0.071
	UMGA		**2.501**	**2.499**	**2.496**	**2.496**	**2.496**	**2.502**	**2.498**	**2.502**	**2.499**	**0.006**
	L&T		2.500	2.484	2.494	2.475	2.469	2.461	2.462	2.486	2.458	0.026
	Chester		2.506	2.486	2.476	2.476	2.456	2.456	2.446	2.436	2.426	0.039
Q	EAGA	0.405	0.404	0.404	0.402	0.395	0.399	0.401	0.392	0.400	0.397	0.005
	UMGA		**0.404**	**0.403**	**0.403**	**0.403**	**0.403**	**0.403**	**0.402**	**0.403**	**0.402**	**0.002**
	L&T		0.402	0.401	0.401	0.396	0.394	0.395	0.389	0.387	0.385	0.010
T	EAGA	0.226	0.224	0.219	0.148	0.130	0.110	0.104	0.086	0.078	0.082	0.085
	UMGA		**0.224**	**0.224**	**0.224**	**0.224**	**0.223**	**0.225**	**0.224**	**0.223**	**0.224**	**0.001**
	L&T		0.225	0.223	0.224	0.221	0.222	0.220	0.219	0.221	0.221	0.004
	Chester		0.219	0.215	0.207	0.205	0.200	0.226	0.190	0.185	0.183	0.020
$SC(\times 10^{29})$	EAGA	1.218	0.472	0.027	0.011	0.003	0.001	0.001	0.009	0.001	0.001	1.044
	UMGA		**1.052**	**0.932**	**1.040**	**1.068**	**0.871**	**0.921**	**0.875**	**0.776**	**0.765**	**0.266**
	L&T		2.730	1.870	3.610	3.400	1.450	6.940	6.250	4.460	4.040	2.386
	Chester		1.300	1.410	2.160	2.880	2.660	5.550	5.370	11.000	8.250	2.969

For each dataset and algorithm, we vary k from 1 to 10 ($k = 1$ correspond to original dataset) and compare the results obtained on λ_1, μ_2, h, Q, T and SC. The last column correspond to the average error $\langle\mathcal{E}\rangle$. Bold rows indicate the algorithm that achieves the best results (i.e, lowest information loss) for each measure. Values of Liu and Terzi algorithm are taken from Ying et al. [35] and values of Chester et al. algorithm are taken from [12]

knowledge, i.e., the adversary tries to re-identify a user in the anonymous data using the degree of some target vertices.

Four relevant methods of k-degree anonymity have been surveyed and compared. They are the algorithm by Liu and Terzi in [26], the approach using evolutionary algorithms and univariate micro-aggregation by Casas-Roma et al. in [6, 8], and the method based on vertex addition instead of only changing the edge set by Chester et al. in [12].

As we have stated before, the best results are achieved by the UMGA algorithm. We point out two important factors in order to reduce the information loss and preserve the data utility. Firstly, it is important to minimise the number of modifications in edge and vertex set, and secondly, considering the edge relevance will reduce the noise in the anonymous data and preserve the structural and spectral properties.

Acknowledgments This work was partly funded by the Spanish Government through projects TIN2011-27076-C03-02 "CO-PRIVACY" and CONSOLIDER INGENIO 2010 CSD2007-0004 "ARES".

References

1. Adamic, L.A., Glance, N.: The political blogosphere and the 2004 U.S. election. In: Internatioanl Workshop on Link Discovery, pp. 36–43. ACM, USA (2005)
2. Backstrom, L., Dwork, C., Kleinberg, J.: Wherefore art thou r3579x? anonymized social networks, hidden patterns, and structural steganography. In: International Conference on World Wide Web, pp. 181–190. ACM, USA (2007)
3. Bhagat, S., Cormode, G., Krishnamurthy, B., Srivastava, D.: Class-based graph anonymization for social network data. Proc. VLDB Endowment **2**(1), 766–777 (2009)
4. Boldi, P., Bonchi, F., Gionis, A., Tassa, T.: Injecting uncertainty in graphs for identity obfuscation. Proc. VLDB Endowment **5**(11), 1376–1387 (2012)
5. Campan, A., Truta, T.M.: Data and structural k-anonymity in social networks. In: Privacy, Security, and Trust in KDD, pp. 33–54. Springer (2009)
6. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: An algorithm For k-degree anonymity on large networks. In: IEEE International Conference on Advances on Social Networks Analysis and Mining, pp. 671–675. IEEE, Niagara Falls (2013)
7. Casas-Roma, J., Herrera-joancomartí, J., Torra, V.: Analyzing the impact of edge modifications on networks. In: International Conference on Modeling Decisions for Artificial Intelligence, pp. 296–307. Springer, Barcelona (2013)
8. Casas-Roma, J., Herrera-Joancomartí, J., Torra, V.: Evolutionary algorithm for graph anonymization (2013). ArXiv:1310.0229v2 [cs.DB], pp. 1–6
9. Cheng, J., Fu, A.W., Liu, J.: K-Isomorphism: privacy preserving network publication against structural attacks. In: International Conference on Management of Data, pp. 459–470. ACM, USA (2010)
10. Chester, S., Kapron, B.M., Ramesh, G., Srivastava, G., Thomo, A., Venkatesh, S.: k-Anonymization of social networks By vertex addition. In: ADBIS 2011 Research Communications, pp. 107–116 (2011)
11. Chester, S., Gaertner, J., Stege, U., Venkatesh, S.: Anonymizing subsets of social networks with degree constrained subgraphs. In: IEEE International Conference on Advances on Social Networks Analysis and Mining, pp. 418–422. Washington, IEEE (2012)
12. Chester, S., Kapron, B.M., Ramesh, G., Srivastava, G., Thomo, A., Venkatesh, S.: Why Waldo befriended the dummy? k-Anonymization of social networks with pseudo-nodes. Soc. Netw. Anal. Min. **3**(3), 381–399 (2013)

13. Chester, S., Kapron, B.M., Srivastava, G., Venkatesh, S.: Complexity of social network anonymization. Soc. Netw. Anal. Min. **3**(2), 151–166 (2013)
14. Cormode, G., Srivastava, D., Yu, T., Zhang, Q.: Anonymizing bipartite graph data using safe groupings. VLDB J. **19**(1), 115–139 (2010)
15. De Capitani di Vimercati, S., Foresti, S., Livraga, G., Samarati, P.: Data privacy: definitions and techniques. Int. J. Fuzziness Knowl. Based Syst. **20**(6), 793–818 (2012)
16. Dwork, C.: Differential privacy. Int. Conf. Automata Lang. Program. **4052**, 1–12 (2006)
17. Hanhijärvi, S., Garriga, G.C., Puolamäki, K.: Randomization techniques for graphs. In: SIAM Conference on Data Mining, pp. 780–791. SIAM, USA (2009)
18. Hansen, S.L., Mukherjee, S.: A polynomial algorithm for optimal univariate microaggregation. IEEE Trans. Knowl. Data Eng. **15**(4), 1043–1044 (2003)
19. Hay, M., Miklau, G., Jensen, D., Weis, P., Srivastava, S.: Anonymizing social networks. Technical Report 07–19, UMass Amherst, pp. 1–17 (2007)
20. Hay, M., Miklau, G., Jensen, D., Towsley, D., Weis, P.: Resisting structural re-identification in anonymized social networks. Proc. VLDB Endowment **1**(1), 102–114 (2008)
21. Hay, M., Liu, K., Miklau, G., Pei, J., Terzi, E.: Privacy-aware data management in information networks. In: International Conference on Management of Data, pp. 1201–1204. ACM, New York (2011)
22. Hay, M., Li, C., Miklau, G., Jensen, D.: Accurate estimation of the degree distribution of private networks. In: IEEE International Conference on Data Mining, pp. 169–178. IEEE, Miami (2009)
23. Kapron, B.M., Srivastava, G., Venkatesh, S.: Social network anonymization via edge addition. In: IEEE International Conference on Advances on Social Networks Analysis and Mining, pp. 155–162. IEEE, Kaohsiung (2011)
24. Krebs, V.: (2006). http://www.orgnet.com
25. Li, N., Li, T., Venkatasubramanian, S.: t-Closeness: privacy beyond k-anonymity and ℓ-Diversity. In: IEEE International Conference on Data Engineering, pp. 106–115. IEEE (2007)
26. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: ACM SIGMOD International Conference on Management of Data, pp. 93–106. ACM, New York (2008)
27. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: ℓ-diversity: privacy beyond k-anonymity. ACM Trans. Knowl. Disc. Data **1**(1), 3:1–3:12 (2007)
28. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: IEEE Symposium on Security and Privacy, pp. 173–187. IEEE, Washington (2009)
29. Sihag, V.K.: A clustering approach for structural k-anonymity in social networks using genetic algorithm. In: CUBE International Information Technology Conference, pp. 701–706. ACM, Pune (2012)
30. Samarati, P.: Protecting respondents identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)
31. Stokes, K., Torra, V.: Reidentification and k-anonymity: a model for disclosure risk in graphs. Soft Comput. **16**(10), 1657–1670 (2012)
32. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertainty Fuzziness Knowl. Based Syst **10**(5), 557–570 (2002)
33. Tripathy, B.K., Panda, G.K.: A new approach to manage security against neighborhood attacks in social networks. In: IEEE International Conference on Advances on Social Networks Analysis and Mining, pp. 264–269. IEEE, Odense (2010)
34. Wu, W., Xiao, Y., Wang, W., He, Z., Wang, Z.: K-symmetry model for identity anonymization in social networks. In: International Conference on Extending Database Technology, pp. 111–122. ACM, USA (2010)
35. Ying, X., Pan, K., Wu, X., Guo, L.: Comparisons of randomization and k-degree anonymization schemes for privacy preserving social network publishing. In: Workshop on Social Network Mining and Analysis, pp. 10:1–10:10. ACM, USA (2009)
36. Ying, X., Wu, X.: Randomizing social networks: a spectrum preserving approach. In: SIAM Confernece on Data Mining, pp. 739–750. SIAM, Atlanta (2008)

37. Ying, X., Wu, X.: Graph generation with prescribed feature constraints. In: SIAM Conference on Data Mining, pp. 966–977. SIAM, Sparks (2009)
38. Zheleva, E., Getoor, L.: Privacy in social networks: a survey. In: Aggarwal, C.C. (ed.) Social Network Data Analytics, pp. 277–306. Springer, New York (2011)
39. Zhou, B., Pei, J.: Preserving privacy in social networks against neighborhood attacks. In: IEEE International Conference on Data Engineering, pp. 506–515. IEEE, USA (2008)
40. Zhou, B., Pei, J.: The k-anonymity and ℓ-diversity approaches for privacy preservation in social networks against neighborhood attacks. Knowl. Inf. Syst. **28**(1), 47–77 (2011)
41. Zou, L., Chen, L., Özsu, M.T.: k-automorphism: a general framework for privacy preserving network publication. Proc. VLDB Endowment **2**(1), 946–957 (2009)

Evaluating Privacy Risks in Social Networks from the User's Perspective

Michal Sramka

Abstract Determining privacy risks when publishing information on social networks often presents a challenge for the users. A measure of how much of sensitive information users shared with others on a social network website would help the users to understand whether they individually share too much. We survey existing measures that evaluate privacy from the user's perspective or help the user with the privacy risks and related decisions in social networks. We present the Privacy Scores—a measurement of how much sensitive information a user made available for others on a social network website, discuss some of their shortcomings, and discuss research directions for their extensions. In particular, we present our proposal for an extension that takes the privacy score metric from a single social network closed system to include auxiliary background knowledge. Our examples and experimental results show the need to include publicly available background knowledge in the computation of privacy scores in order to get scores that reflect the privacy risks of the users more truthfully. We add background knowledge about users by means of combining several social networks together or by using simple web search for detecting publicly known information about the evaluated users. This is a revision and extension of our former paper.

1 Introduction

Recently there was an explosion of popularity of web sites that allow users to share information. These sites—social-network sites, blogs, and forums such as Google+, Facebook, LinkedIn and others—attract millions of users. The users publish and share information about themselves by creating online profiles, posting blogs and comments. Such information usually contains personal details. Often the users are

M. Sramka (✉)
Faculty of Electrical Engineering and Information Technology, Institute of Computer Science and Mathematics, Slovak University of Technology, Ilkovičova 3, 812 17 Bratislava, Slovakia
e-mail: sramka@stuba.sk

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_14

unaware of the potential risks involved when they are sharing sensitive information online. Quantifying the individuals' privacy risk due to these information-sharing activities of the individuals is a challenging task. Yet the users should know where they stand on the privacy measuring scale.

Securing individuals' privacy in such environments and protecting users against threats such as *identity theft*. *Digital stalking* or *cyberstalking* becomes an increasingly important issue. Both users and service providers recognize the need for users' privacy. The sites may provide some privacy controls. However, the users are faced with too many options and too many controls, and lack the understanding of privacy risks and threats or are unable to accurately asses them. This all contributes to the confusion for the average users, and often results in skipping the complicated and time-consuming tasks of setting the privacy controls that should protect them.

It needs to be noted that there are research directions that try to help the social network users by enabling them to set and personalize their online privacy preferences automatically [1]. But even with properly configured privacy settings for a user profile, some privacy concerns remain. Take for example discussion forums, where tenths or hundreds contributions to multiple discussions of various topics are written by a user. Although the user is careful not to disclose any personally identifiable information in his/her individual posts, personal, sensitive, and private information may be inferred and disclosed by looking at the set of all posts by the user. From the cumulative set of all posts, it may be then possible to profile the user and infer the user's opinions or even identity.

There are primarily two privacy issues [2] in social networks. A lot of research exists dealing with the privacy concerns of publishing the social network data without revealing the identity of an individual. The other privacy risk in social networks comes from the information that has been shared by the users on their profiles:

- *Relationship privacy.* Generally, a social network consist of users and relationships among them. The relationships can be of different kinds—such as "colleague of", "friend of", etc.—and of different trust level—for example, direct relationship, friend-of-a-friend. The availability of information on relationships raises privacy concerns: Knowing who is trusted by whom and to what extent discloses information about the users, their thoughts and feelings. Sometimes just the fact that a relationship exists can be a privacy leakage.
- *Content privacy.* The information content a user shares or publishes on a social network clearly affects the user's privacy. The user can share sensitive or personal information with his/her friends, their friends, or using similar schemed up to sharing completely, that is, basically publishing the information for all. Often some information about the user that s/he wants to keep private can be inferred from other shared information or from information shared with other users.

For a survey of privacy research in social networks see [3], more references are in Sect. 5. Here, we are concerned with the privacy risks from the user's perspective. That is, we focus on measuring the privacy of social network users and helping and enabling them to make informed decisions about their sharing activities, following the research direction of [1, 4–6].

2 Privacy from the User's Point of View

We focus on privacy from the user's point of view. We survey some existing models and measures of user's privacy that empower the users by providing immediate decision support about their actions and their impact on the user's privacy.

Orthogonal to the measures are the tools that help the social network users make informed and wise choices about their privacy settings. We also briefly describe some of these tools.

2.1 Privacy Scores

Privacy Scores by Liu and Terzi [4, 5] were proposed to quantify the privacy risks of individuals posed by their profiles in a social-network site. Focus here is on privacy risks from the individuals' perspective. In the proposed framework, each user in a social network is assigned a privacy score based on the information in his/her profile compared to all available information in all profiles. The score then measures the user's potential privacy risk due to having his/her profile available on the social-network site.

The main drawbacks of this proposal of privacy scores are the concentration only on users' profiles and inconsideration of other publicly available information about the users on the same social network and beyond it. In particular, *background knowledge* about a user is not included in the computation of the privacy score. Background knowledge is some information about an individual that by itself is not a privacy disclosure, but combined with other information it becomes one. Background knowledge is sometimes referred to as external knowledge or auxiliary information.

The value of a Privacy Score is a combination of each one of user's profile items, labelled $1, \ldots, n$, for example, real name, email, hometown, land line number, cell phone number, relationship status, IM screen name, etc. The contribution of each profile item to privacy score is based on sensitivity and visibility. The *sensitivity* β_i depends on the item i itself—the more sensitive the item is, the higher is the privacy risk of it being revealed. The *visibility* of an item i belonging to a user j is denoted $V(i, j)$ and captures how far this item is known in the network—the wider the spread in the network, the higher the visibility.

The privacy score of an item i belonging to a user j is simply $\text{PR}(i, j) = \beta_i \times V(i, j)$. The overall *privacy score* for a user j with n items is then computed as

$$\text{PR}(j) = \sum_{i=1}^{n} \text{PR}(i, j) = \sum_{i=1}^{n} \beta_i \times V(i, j) \ . \tag{1}$$

To keep the privacy score PR a non-decreasing function, in order for it to be a nicely behaving score, both the sensitivity β_i and visibility $V(i, j)$ must be non-negative functions. In practice, the sensitivity and visibility are determined from an $n \times m$

matrix R that represents n items for m users of a single social network. The value of each cell $R(i, j)$ describes the willingness of the user j to disclose the item i. In the simplest case, the value of $R(i, j)$ is 0 if the user j made the item i private and 1 if the item i is made publicly available. From this, the (observed) visibility can be defined as $V(i, j) = R(i, j)$. In a more granular approach, the matrix R can be defined by $R(i, j) = k$, representing that the user j disclosed the item i to all the users that are at most k jumps away in the graph of the social network. Regarding the sensitivity of an item, β_i can be computed using Item Response Theory (IRT) [4, 5]. The IRT can be also used to compute the true visibility of an item for a user.

The privacy score is computed for each user individually. It is an indicator of the user's potential privacy risk—the higher the score of a user, the higher the threat to his/her privacy.

2.2 Privacy Quotient and Privacy Armor

One extension of Privacy Scores comes under the name of *Privacy Quotient* [7]. The authors, similar to our past research [8], have realized that unstructured data pose a problem for privacy score evaluation. The focus here is to evaluate a user's privacy risks in exchanging, sharing, publishing, and disclosing unstructured data—namely, text messages.

A (text) message may contain sensitive information about the user. The message is first checked for any sensitive information such as the user's phone numbers, address, email, or location. The message is then classified as sensitive or non-sensitive by means of a naive binary classifier.

Each sensitive part of the message is treated as an "item" that has some sensitivity. The Privacy Quotient computation is then the same as for the privacy scores, that is, using the Eq. (1). In addition, the message's privacy leakage ϑ is computed as the ratio of sum of all the sensitive parts(items) sensitivities β_i to the sum of all the sensitivities. This privacy leakage ϑ is similar to the computation of the Privacy Index PIDX discussed next in Sect. 2.4 and the Eq. (3).

The authors of Privacy Quotient also proposed the *Privacy Armor* model: For any message a user wants to share with his/her group of friends, the quotient (=score) is computed for not just the message, but an average quotient is computed for the whole group of friends. If the average quotient of the group is above some threshold—some fixed desired quotient value, an alert containing the message's privacy leak may be present to the user, and the message may be anonymized before being sent to the group.

2.3 Privacy-Functionality Score

An interesting research direction motivated by the Privacy Score is the Privacy-Functionality Score [2]. A utility function based on the original privacy scores is

proposed. The utility function measures the rational benefit derived by a user from his/her participation in a social network, in the terms of information acquired versus information provided. The utility is defined as the functionality the user gets divided by privacy risk score the user incurs, that is, the amount of information the user can see about other users in the social network divided by the amount of information the user reveals about himself/herself. The Privacy-Functionality Score of user j, using the notations from Sect. 2.1 and the Eq. (1), is

$$\mathrm{PRF}(j) = \frac{\sum_{j'=1, j' \neq j}^{n} \mathrm{PR}(j')}{1 + \mathrm{PR}(j)} = \frac{\sum_{j'=1, j' \neq j}^{n} \sum_{i=1}^{n} \beta_i \times V(i, j')}{1 + \sum_{i=1}^{n} \beta_i \times V(i, j)} \ . \tag{2}$$

Using this score and considering the social network and privacy to be a game where users are players, the author was able to derive two results.

The first result is when users of a social network try to selfishly maximize this utility score—that is, the users are "free riding" the social network by offering and sharing no information about themselves and only acquiring information from other users. If each user of the social network is independently choosing this strategy, then this case results in the non-functionality and shutdown of the social network.

The second result is based on a game where users choose correlated strategies to jointly get the maximum utility score from the social network. Such strategy indeed exists—the simplest one being "tit-for-tat", where items are disclosed among users sequentially: A user starts the round by revealing the least sensitive item i that has not been shared yet. A next round, where more sensitive items are disclosed, does not start unless all users in a group or the whole social network have revealed the item i. This strategy or a similar reputation-based strategy [2] can be used to assist users in making rational decisions regarding which of his/her attributes the user reveals to other users in a social network.

2.4 PIDX

Privacy Index [9], PIDX, is used to describe a user's privacy exposure factor based on the known (published/shared, in our terminology) attributes. Higher PIDX indicates higher exposure of privacy. PIDX as the proposed privacy risk indicator can be calculated in real time and the value can be used for privacy monitoring and risk control, same as is the case with the previously discussed metrics.

PIDX is defined as the ratio of the sum of the privacy impact factors of the published items, set K, to the sum of the the privacy impact factor of all the items, set I. That is,

$$\mathrm{PIDX} = \frac{\sum_{k \in K} s_k}{\sum_{i \in I} s_i} \times 100 \ , \tag{3}$$

where s_i are privacy impact factors of an item i defined as the sensitivity of the item i, assuming the visibility of the published items to be 1. Since $K \subseteq I$, it is

obvious that Privacy Index PIDX is a score between 0 and 100 and reflects how much sensitive information has the user published. In this sense, the Privacy Index PIDX computation for the user is the same as for the computation of a messages's privacy leakage ϑ of Privacy Quotient, discussed in Sect. 2.2, because sensitivity of an item i is $s_i = \beta_i$.

The authors use the privacy index in a model for privacy ranking and monitoring that employs web searching to look for already known and published items from a user. The web searching can use standard search engines as well as it can be based on the deep web search engines. This is similar to our approach [8].

2.5 PrivAware

Although not a score or metric, PrivAware [6] is a tool to detect and report unintended information loss in social networks. The authors propose to quantify and reduce privacy risks attributed to friends in online social networks. PrivAware tool provides two functions—inference detection and inference reduction.

First, PrivAware infers the attributes (items) of a user based on those of his/her friends. In particular, the tool tries to detect whether attributes of the user at hand can be inferred given all the attributes of his/her friends. PrivAware derives inferences for the following attributes: age, country, state, zip, high school name, high school grad year, university, degree, employer, affiliation, relationship status, and political view.

Second, PrivAware suggests how to change the members of the user's friends to reduce the number of inferable attributes to an acceptable level. The user can simply, but drastically, cut the relationships to his/her friends in order to remove the inferences, or the user can configure his/her privacy settings for these friends in more stringent manner.

2.6 Privometer

The authors in [10] develop a privacy-protection tool, Privometer, to measure the amount of sensitive information leakage in a user's profile. The leakage is indicated by a numerical value. The tool can suggest self-sanitization actions based on the numerical value.

The importance of the research this tool introduced is in looking beyond the publicly available information that the user shares on his/her profile. The model of Privometer also considers substantially more information that a potentially malicious application installed in the user's friend realm can access. Of course, this only applies to social network websites, such as Facebook, that allows applications to access users' information.

2.7 Tools for Social Network Privacy Settings Configuration

As discussed, the social network websites usually provide some privacy controls in the form of a settings page. However, the users are faced with too many options and too many controls, and lack the understanding of privacy risks and threats or are unable to accurately asses them. This all contributes to the confusion for the average users, and often results in skipping the complicated and time-consuming tasks of setting the privacy controls that should protect them.

Here we briefly describe some of the research tools that help the social network users make informed and wise choices about their privacy settings.

2.7.1 Privacy Wizard

Considering this problem of privacy settings, the authors of [1] have proposed a template of a social networking privacy wizard. The idea of the Privacy Wizard is that from a set of user's privacy choices in the form of rules, it is possible to design and build a machine learning model. Such model can then be used to configure the user's privacy settings automatically.

2.7.2 PViz

Another tool for configuring the user's privacy settings is PViz [11]. PViz tool is centered on a graphical display of the privacy choices. It allows the user to understand the visibility of his/her profile according to natural sub-groupings of friends, and at different levels of granularity.

3 Assessing Privacy Risks Beyond Social Networks

Here follows our contribution to the area of assessing and evaluating privacy risks of users in social networks pertained from publishing possibly sensitive information about themselves. This part follows our original research [8].

3.1 Our Contribution

We propose a new concept for Privacy Scores that were introduced in Sect. 2.1. We explore the idea of presenting users with a new privacy score that measures their overall potential privacy risk due to available public information about them. Compared with the original Privacy Scores by Liu and Terzi [4], we overcome the

drawbacks of concentrating only on users' profiles in a single social network, and we include publicly available background knowledge in computation of the new privacy scores. Our new privacy scores metric better represents the potential privacy risks of users and thus helps them make better decision in managing their privacy.

Our results are twofold. Firstly, in Sect. 3.2 we discuss the shortcomings of the privacy scores. We present several opportunities for extending the original privacy scores. With the extension of including background knowledge in mind, we identify some background knowledge that is publicly available but that cannot be easily extracted by computers in an automatic manner. Secondly, we proposed an extension of the privacy score metric that takes it from a closed system evaluating privacy over a single social network to a metric that includes information about the users that comes from outside the social network. In Sect. 3.3, we present examples and experimental results showing paradoxes that may happen when the computation is over only a single social network. Next, in Sect. 4, we extend the computation of privacy scores to include two or multiple social networks. Our final proposal, in Sect. 4.2, uses web searches to include all available public indexed human knowledge in the computation of the privacy score of a user. Thus, our new privacy score reflects the privacy risks of combining user's profile information with available knowledge about the user represented by the web.

Our proposed method for making web search inferences while scoring the privacy risks of individuals can also be seen as a privacy attack. However, we do not explore this direction, as there are already too many attacks, some of them referenced later in Sect. 5. Our contribution rather focuses on helping users achieve their privacy needs and lower their privacy risks. The extended privacy score helps the users to make more informed decisions about their online activities.

3.2 Shortcomings and Opportunities of Privacy Scores

The privacy score, presented in Sect. 2.1, is no doubt a useful metric for each and every user of a social network. Nevertheless, there exist several shortcomings of the originally proposed privacy scores. We list a few of them here. Some of these were already noticed and identified by the authors of the privacy scores, others are just observations, and some are our proposals for further exploration, research, and enhancements of privacy scores.

Regarding the items of a user profile, one can immediately notice hardship in quantifying the items themselves:

- The granularity of profile items is of particular concern. For example, the profile item "personal hobbies" can cover a range of non-private and private information and so its true sensitivity cannot be really established for the general case required by the privacy scores.
- Different profile items have different life-cycles. Some profile items may have a time attribute attached to them—for example, a cell/mobile phone number

or an address are temporary information, while the date of birth or the mother's maiden name are permanent for life. The proposed privacy score, as defined, ignores these facts. We believe that implicit time relevance should be taken into account for more precise evaluation of a user's privacy.

- Impossibility or hardness of including all, possibly private, information in privacy score computation. For example, consider photos: It may be hard or impossible in some cases to (automatically or even by a human involvement/assessment) establish relationships from photos. Or whether a person is drinking alcohol in a photo. Another example are discussion forums: Information is exhibited in natural language form. Determining a political orientation of a user from a single post may not be possible, yet looking at the cumulative set of the user's posts, private information can be inferred about the user (see Sect. 3.3).

Of more concern and interest is the definition, computation and use of sensitivity β_i for item i. As proposed in Privacy Scores, the sensitivity is computed from the matrix R, that is, the sensitivity is based only on the users and items in the single social network. When considering a single social network represented by a matrix R, it is easy to get a wrong perception of privacy due to the limited information about the users.

- The sensitivity β_i computed for an item i would reflect the true real-world sensitivity of this item only if the distribution of people in the social network would mirror the real-world distribution. Obviously, many social networks are not like this, and so paradoxes are likely because of this fact. For example, take a date of birth that most people consider a sensitive and private information. However, if everybody in a social network reveals his/her date of birth, then this item will be considered as not sensitive at all (because everybody reveals it). Paradoxes on the other side of the spectrum are possible, too. For example, if an item in a social network is filled only by one or a few users, because the other users are too lazy to fill it in, then the item will be considered sensitive (by the computation of sensitivity), although the item is far from being considered sensitive or private in the real life. For this reasons, the definition of the sensitivity is not the best possible as it does consider only published information and not the true perception of privacy of the users.
- No background knowledge inclusion, and so no inference detection or control: A privacy metric should include "background knowledge" (auxiliary information or external knowledge) in establishing a score for a user. Speaking more generally, a single social network or any closed system evaluation is not sufficient for real and proper privacy evaluation of a user.
  For privacy scores, this means that the computation of the score should not depend only on the matrix R coming from a single social network. Several extensions of the original privacy score metric are possible based on the background knowledge type and source. In Sect. 4 we propose a new method to compute privacy scores, one that considers information about users beyond the ones in the social network, namely from a second/other social networks or more generally from the web.

Finally, it needs to be mentioned that the proposed privacy score metric measures only some aspect of privacy, namely attribute (item) disclosure and identity disclosure

arising from the attribute disclosure. There are several other aspects that may be of concern to the users of a social network, such as:

- the risks of identity disclosure that is not based on attribute disclosure—for example, based on behavioral observations,
- the risk of identity theft,
- the risk of link or relationship disclosure,
- the risk of group membership disclosure, or
- the risk of digital stalking.

How to measure these risks and help the users making informed decisions by presenting them a score reflecting these risks is still an open problem.

3.3 A Discussion Forum

The computation of privacy scores proposed by Liu and Terzi [4, 5] introduced in Sect. 2.1 assumes the analyzed information to be readily available for inclusion in the matrix R. As we noted in Sect. 3.2, non-structured information cannot be always easily included for analysis. It may be either information that is hard to extract—for example, relationships from photos—or previously not defined information—for example, non-structured text in natural language may contain multiple private items some of which may not be pre-defined as items of the matrix R.

Together with my Master's student Ján Žbirka we performed a few experiments [12], where simple natural language analysis was used to determine if some private information has been included in discussion comments on a news website. Since the users usually post multiple comments, they may contain multiple private information that must be looked-up for inclusion in the privacy scores. In our experiments, shown in the next section, we concentrated on information about political orientation before election and religious believes.

3.3.1 Experimental Results

Discussions of the Slovak news web site www.sme.sk were analyzed just before the government election in March 2012. From all the users that posted comments on the website, 5,268 users who posted more than 500 comments over the lifetime of the website were considered as the most active ones. In the three weeks before the election, these 5,268 most-active users posted 43,035 comments that were analyzed. Almost 20 % of the analyzed users revealed in their comments which political party in particular they were or were not going to vote for.

Summary of the findings are in Table 1 and all the other details about the experiment can be found in the Master's thesis of Ján Žbirka [12].

Since discussions on this website about religion and church are very heated, we also analyzed whether it is possible to find out the faith/religious beliefs of the users from their comments. The experiment that was done on the same sample of the users and comments have shown that simple natural language analysis can determine

Table 1 The number of the users (from the total of 5,268) who were find to disclose this information in discussion comments

Users who will	Vote	Not vote
At all	763 (14.5 %)	173 (3.3 %)
For a right wing party	209 (4.0 %)	194 (3.7 %)
For a left wing party	59 (1.1 %)	46 (0.9 %)
For a particular party	688 (13.1 %)	335 (6.4 %)

faith, although the users were more conservative in revealing their religious believes compared to the political orientation. In total, 133 (2.5 %) users were found to disclose their religion, and 106 (2.0 %) users were found to disclose that they are atheists.

4 Privacy Score Extension

The biggest disadvantage of the privacy scores that were outlined in Sect. 2.1 is the non-consideration of background knowledge. *Background knowledge* (sometimes referred to as external knowledge or auxiliary information) is some information about an individual that by itself is not a privacy disclosure, but combined with other information it becomes one. We propose two possible extensions of the original privacy score metric that take public background knowledge into account.

It needs to be noted that the reason to include background knowledge in the computation of the privacy score is two-fold. On the one hand, such extended privacy score will more precisely present users with privacy risks arising from publishing their information. On the other hand, using background knowledge also reduces another shortcoming of the original privacy scores. Namely, the more background knowledge is considered, the closer is the sensitivity of items to the true sensitivity. In other words, adding background knowledge to the privacy score computation also reduces or eliminates sensitivity paradoxes—see Sect. 3.2.

Our extended privacy score metric uses the same formula as in the Eq. (1) with sensitivity and visibility as the original privacy scores, but the information that is used to compute these—the matrix R—is extended by additional knowledge. We discuss two instances of this extension. The first one, presented next, combines information from two or several social networks when evaluating privacy risk of a user. The second instantiation of the extended privacy score metric, which we present in Sect. 4.2, uses "all the human knowledge" in privacy risk evaluation.

Our proposal of a simple inclusion of additional information in the privacy score computation is based on users' information (items) from multiple social networks. Let N be the number of considered social networks, and let R_t be as the already defined matrix R for a social network t, with $t = 1, \ldots, N$. Hence, R_t is a $n \times m$ matrix, where $R_t(i, j)$ represents the publicity of an item i for a user j—that is, non-disclosure when $R_t(i, j) = 0$ or disclosure when $R_t(i, j) > 0$ and possibly how far from the user j is the item public in the (graph of the) social network t.

It is likely in practice that not all the users are in every social network and that every item is in each of the corresponding profiles. Here we assume that the range of the items $i = 1, \ldots, n$ and the range of the users $j = 1, \ldots, m$ are the supersets over all the social networks, and so $R_t(i, j) = 0$ if an item i or user j do not exist in the social network t. We define the matrix R used for sensitivity and visibility computation as $R(i, j) = \max_t R_t(i, j)$ and use the formula from the Eq. (1) to compute the privacy score. Such privacy score better estimates the risk of privacy disclosure.

Together with my Master's student Lucia Maringová we performed a few experiments [13], where the same users on two social networks were evaluated for their privacy risks. The two social networks were of different type, so it was expected that the users would behave differently and therefore would disclose different amount of information about themselves on each social network. In our experiments, shown in the next section, we focused on computing privacy scores from each social network individually and then comparing the behavior of people in the terms of private information disclosure on two social networks.

4.1 Experimental Results

The purpose of the experiment is to show that privacy risks, as measured by the original and extended privacy score, are higher when two social networks are combined. Specifically, this means that some users tend to be conservative in one social network while publicly disclose private information in another social network.

For the experiments, profiles from the same users on two social networks were downloaded and analyzed. The social networks (websites) were Pokec.sk and Zoznamka.sk. They both belong to the same content provider/operator, and so use the same user authentication, which facilitated the pairing of the users from the two networks. Zoznamka.sk is a dating website, where a profile can contain up to 5 items: age, body type, weight, height, and contact. Pokec.sk is a website about chatting, messaging, and picture sharing. A profile on Pokec.sk can contain up to 34 items.

A sample of 3,923 users was selected. From all these users, there are only 23 users ($<1\,\%$) who completely filled all profile items on both websites. These people probably do not understand the risks of disclosing private information or ignore these risks, whether consciously or unconsciously by making a mistake. Roughly 32% of the users shared the same information on both sites.

Because of the nature of the website, users on the dating website Zoznamka.sk revealed more personal information about themselves. This is likely due to the fact that the users tried to create interest and attract the users who viewed their profiles. No user had less than 2 items (out of 5) filled on Zoznamka.sk. Conversely, many users on Pokec.sk left their profiles empty. What is of interest to us are the users who had empty profiles on Pokec.sk and non-empty profiles on Zoznamka.sk. Table 2 summarizes these users. All the details can be found in the Master's thesis of Lucia Maringová [13].

Table 2 The number of users who shared nothing on Pokec.sk, but had non-empty profiles on Zoznamka.sk Note that minimum items filled in on Zoznamka.sk was 2

On Pokec.sk	On Zoznamka.sk	# of users
0 items	2 items	542
0 items	3 items	107
0 items	4 items	961
0 items	5 items	119
0 items	>0 items	1,727
		44 %

In the terms of the privacy score, the users on Pokec.sk who had empty profiles would receive the score of 0, because they do not share or disclose anything. However, this would be awfully wrong in any privacy risk analysis, because private information about these users is publicly available and linkable to these users. At least two additional items can be learned about roughly 44 % of the users with empty profiles on Pokec.sk when considering Zoznamka.sk, so the extended privacy score computed over both networks for these users will be non-zero. This simple experiment itself shows the need to extend the original privacy scores from analyzing information over one social network to analyzing also auxiliary information.

4.2 Using All the Human Knowledge in Privacy Score Computation

Extending the original Privacy Scores by Liu and Terzi [4, 5] to multiple social networks certainly helps in privacy risk evaluation. The selection of social networks included in the extended privacy score computation, presented above, strongly impacts the quality and truthfulness of the score. The most truthful privacy risk evaluation can be achieved if all the human knowledge is used in the computation of the privacy score.

Including "all the human knowledge" in any computation is obviously impossible, so an approximation would have to suffice for all practical purposes. To effectively include the knowledge, we need to be able to quickly search for particular information or relation. Thus, we should use all the *indexed* human knowledge. Private databases and the "deep web" are believed to contain much more information than what is publicly available. In general, private information is out of reach for privacy adversaries as well as for privacy evaluators. Hence, we foresee to use all the *indexed public* human knowledge in the privacy score computation. Currently, the best instance and the best source of all the indexed public human knowledge is (Google) web search. In fact, there exists a proposal, namely Web-Based Inference Detection [14, 15], which takes advantage of the assumption that the web search is the proxy for all human knowledge.

Our idea is as follows: If an item of a user is not disclosed in the social network, we want to determine if the item has been disclosed elsewhere by using an inference

detection based on the other disclosed items for the user. Our inference detection method is heavily influenced by the Web-Based Inference Detection [14]. So, our idea rewritten in the terms of inference detection is: If there is a privacy-impacting inference detected for an undisclosed item, then this detected inference should be included in the privacy score computation.

More formally, we propose the following method to compute the privacy score:

Consider a social network of m users each having a possibility to fill a profile of n items. Let R be, as before, the $n \times m$ matrix over $\{0, 1\}$ with $R(i, j)$ representing whether the user j has (or has not) disclosed the item i. Let P be $n \times m$ array of strings with $P(i, j)$ being the value of the item i for the user j, in case this value has been disclosed. Let the set D_i be the domain of the item i. Finally, let β, γ, and δ be positive integers, where β and γ are parameters of the proposed algorithm that control the search depth, and δ is the parameter that controls the number of the most frequent words to be considered. Then the algorithm to extend R and determine the users' disclosures outside the social network is as follows:

For each user j, $j \in \{1, \ldots, m\}$

1. Let $S_j = \{k \mid R(k, j) = 1,\ k = 1, \ldots, n\}$ be the set of all disclosed items for the user j.
2. For each undisclosed item i, that is, for all $i \in \{1, \ldots, n\}$ with $R(i, j) = 0$
   (a) Let T be an empty multiset.
   (b) Take every subset $S'_j \subseteq S_j$ of size $|S'_j| \leq \beta$.
   (c) For every such subset $S'_j = \{i_1, \ldots, i_\ell\}$ with $\ell \leq \beta$
      (i) Use a web search engine to search for keywords $P(i_1, j), \ldots, P(i_\ell, j)$
      (ii) Retrieve the top γ most relevant documents containing these keywords
      (iii) Extract the top δ most frequent words from all these γ documents
      (iv) Add the top δ most frequent words to T together with their frequencies
   (d) Take the most frequent word from T that is also in D_i, if it exists.
   (e) If there is such word, let $R(i, j) = 1$.

After this, the newly enhanced matrix R contains the users' disclosures not just from the social network itself, but also from the web. Visibility and sensitivity values can be then computed from this matrix R, and the privacy score can be computed for each user using the Eq. (1).

The parameters β, γ, and δ can be tuned to achieve different trade-offs between the running time of the algorithm and the completeness and quality of the disclosure detection. In fact, these values can be different for different users, perhaps based on the number of items disclosed in the social network. Additional tuning can be achieved by performing the steps of the algorithm only for those users that have disclosed a "sufficient" number of items in the profile that would allow the web search to identify additional items.

5 Related Work

Our work was influenced by the approach by Liu and Terzi [4, 5], which provides users with a quantification of privacy risks due to sharing their profiles in a social network. Each user is assigned a privacy score based on their and all other users' profile items. The proposal is for a single social-network site, that is, a closed system evaluation of privacy that lacks the consideration and inclusion of background knowledge in computation of the privacy scores. We overcome this shortcoming by including background knowledge in the computation of privacy scores, see Sects. 3.3 and 4.

Privacy Scores is just one metric to help users understand their privacy risks. In Sect. 2, we have surveyed few other measures, scores, and tools—namely, Privacy Quotient and Privacy Armor [7], Privacy-Functionality Score [2], Privacy Index PIDX [9], PrivAware [6], Privometer [10], Privacy Wizard [1], and PViz [11].

The privacy risks of social-network sites are summarized in [16] and more recently in a survey [3]. Several papers present (relationship) privacy attacks in social networks [6, 17–21] or try to lower privacy risks and prevent privacy attacks in social networks [22, 23]. In addition, there are privacy risks from being tracked while browsing these websites [24].

Some form of background knowledge is usually considered in privacy attacks and is very likely available to attackers. Absolute privacy is impossible, because there will be always some background knowledge [25]. Inference techniques can then be used to attack or to help protect private data. In particular, web-based inference detection [14, 15] has been used to redact documents and prevent privacy leaks.

6 Conclusions

As more and more users are joining and using social-network web sites, they become more heavily used and their owners look for new ways to share different content, including private information and information that may lead to unwanted privacy leakages. It becomes increasingly difficult for individuals to control and manage their privacy in the vast amount of information available and collected about them.

Metrics, scores, and tools were proposed to facilitate social network users with a view of their privacy risks. In particular, Privacy Scores is a metric that presents users with a score that reflects their privacy risks arising from disclosing information in their profiles on a social network. We presented several shortcomings of the privacy scores as research opportunities for extending the privacy score metric. Next, we supported the need for extensions by experimental results from different websites and social networks. Finally, we proposed two extensions of the privacy score metric that consider additional background information about the users in the computation of the scores. Our approach provides a better decision support for individuals than the original privacy scores. Based on our extended privacy score metric, the users can compare their privacy risks with other fellow individuals and make informed

decisions about whether they share too much potentially private and sensitive information.

Acknowledgments This work started while the author was with the Universitat Rovira i Virgili, Catalonia and was partly funded by the Spanish Government through projects TSI2007- 65406-C03-01 "E-AEGI" and CONSOLIDER INGENIO 2010 CSD2007-00004 "ARES", and by the Government of Catalonia through grant 2009 SGR 1135. This work was also partly funded by the Slovak grant VEGA 1/0173/13 while the author was with the Slovak University of Technology. Final thanks go to my former Master's students Lucia Maringová and Ján Žbirka for carrying out the experiments.

References

1. Fang, L., LeFevre, K.: Privacy wizards for social networking sites. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 351–360. ACM (2010)
2. Domingo-Ferrer, J.: Rational privacy disclosure in social networks. In: Proceedings of the 7th International Conference on Modeling Decisions for Artificial Intelligence, MDAI 2010, pp. 255–265. Springer (2010)
3. Zheleva, E., Getoor, L.: Privacy in Social Networks: A Survey. In: Social Network Data Analytics, pp. 277–306. Springer (2011)
4. Liu, K., Terzi, E.: A Framework for computing the privacy scores of users in online social networks. In: Proceedings of the Ninth IEEE International Conference on Data Mining, ICDM 2009, IEEE pp. 288–297. (2009)
5. Liu, K., Terzi, E.: A framework for computing the privacy scores of users in online social networks. ACM Trans. Knowl. Discov. Data **5**(1) (2010). (article no.6.)
6. Becker, J., Chen, H.: Measuring privacy risk in online social networks. In: Web 2.0 Security & Privacy 2009 Workshop of 2009 IEEE Symposium on Security and Privacy, W2SP 2009, IEEE (2009)
7. Srivastava, A., Geethakumari, G.: Measuring privacy leaks in online social networks. In: Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2013, pp. 2095–2100. (2013)
8. Sramka, M.: Privacy scores: assessing privacy risks beyond social networks. Infocommunications J. **4**(4), 36–41 (2012)
9. Nepali, R.K., Wang, Y.: Sonet: A social network model for privacy monitoring and ranking. In: Proceedings of the 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops, IEEE, pp. 162–166. (2013)
10. Talukder, N., Ouzzani, M., Elmagarmid, A.K., Elmeleegy, H., Yakout, M.: Privometer: privacy protection in social networks. In: Proceedings of the 2010 IEEE 26th International Conference on Data Engineering Workshops, ICDEW 2010, IEEE, pp. 266–269. (2010)
11. Mazzia, A., LeFevre, K., Adar, E.: Pviz comprehension tool for social network privacy settings. In: Proceedings of the 8th Symposium on Usable Privacy and Security. ACM (2012) (article no. 13.)
12. Zbirka, J.: Privacy risks arising from publishing private information on the web (in Slovak). Master's thesis, Advisor: Michal Sramka, Slovak University of Technology (2012)
13. Maringova, L.: Privacy risks arising from publishing private information in social networks (in Slovak). Master's thesis, Advisor: Michal Sramka, Slovak University of Technology (2012)
14. Staddon, J., Golle, P., Zimny, B.: Web-based inference detection. In: Proceedings of the 2007 USENIX Annual Technical Conference, USENIX 2007, USENIX Association, pp. 71–86. (2007)

15. Chow, R., Golle, P., Staddon, J.: Inference detection technology for Web 2.0. In: Web 2.0 Security & Privacy 2007 Workshop of 2007 IEEE Symposium on Security and Privacy, W2SP 2007, IEEE (2007)
16. Rosenblum, D.S.: What anyone can know: the privacy risks of social networking sites. IEEE Secur. Priv. **5**(3), 40–49 (2007)
17. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: Proceedings of the 30th IEEE Symposium on Security and Privacy, S&P 2009, IEEE, pp. 173–187. (2009)
18. Krishnamurthy, B., Wills, C.E.: On the leakage of personally identifiable information via online social networks. In: Proceedings of the 2nd ACM Workshop on Online Social Networks, WOSN 2009, ACM, pp. 7–12. (2009)
19. Zheleva, E., Getoor, L.: To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In: Proceedings of the 18th International Conference on World Wide Web, WWW 2009, ACM, pp. 531–540. (2009)
20. Korolova, A., Motwani, R., Nabar, S.U., Xu, Y.: Link privacy in social networks. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, ACM, pp. 289–298. (2008)
21. Backstrom, L., Dwork, C., Kleinberg, J.M.: Wherefore art thou R3579X?: anonymized social networks, hidden patterns, and structural steganography. In: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, ACM, pp. 181–190. (2007)
22. Felt, A., Evans, D.: Privacy protection for social networking platforms. In: Web 2.0 Security & Privacy 2008 Workshop of 2008 IEEE Symposium on Security and Privacy, W2SP 2008, IEEE (2008)
23. Zheleva, E., Getoor, L.: Preserving the privacy of sensitive relationships in graph data. In: Proceedings of the First ACM SIGKDD International Workshop on Privacy, Security, and Trust in KDD, PinKDD 2007, ACM, pp. 153–171. (2007)
24. McKinley, K.: Cleaning up after cookies. Technical report, iSEC Partners (2008)
25. Dwork, C.: Differential privacy. In: Proceedings of the 33rd International Colloquium on Automata, Languages and Programming, ICALP 2006, pp. 1–12. Springer (2006)

Part VI
Respondent Privacy: Other Respondent Privacy Enhancing Technologies

Trustworthy Video Surveillance: An Approach Based on Guaranteeing Data Privacy

Antoni Martínez-Ballesté, Agusti Solanas and Hatem A. Rashwan

Abstract Thousands of video files are stored in surveillance databases. Pictures of individuals are considered personal data and, thus, their disclosure must be prevented. Although video surveillance is done for the sake of security, the privacy of individuals could be endangered if the proper measures are not taken. In this chapter we claim that a video-surveillance system could protect our safety and, at the same time, guarantee our privacy. Most literature on privacy in video surveillance systems concentrates on the goal of detecting faces and other regions of interest, and in proposing different methods to protect them. However, the trustworthiness of those systems and, by extension, of the privacy they provide are neglected. Hence, we define the concept of *Trustworthy Video Surveillance System* (T-VSS), which tackles the issue of protecting the privacy of the individuals. In this chapter, we assess the techniques proposed in the literature according to their suitability in a T-VSS. Moreover, we describe a privacy-aware video-surveillance platform that fulfils those properties and we detail all its components. We have implemented and tested the proposed platform to show the feasibility of our proposal.

This work was partly funded by the Spanish Government through project CONSOLIDER INGENIO 2010 CSD2007-0004 ARES, project TIN2011-27076-C03-01 CO-PRIVACY, and by the Rovira i Virgili University through project 2012R2B-01 VIPP.

A. Martínez-Ballesté (✉) · A. Solanas · H.A. Rashwan
Department of Computer Engineering and Maths, Universitat Rovira I Virgili, Tarragona, Spain
e-mail: antoni.martinez@urv.cat

A. Solanas
e-mail: agusti.solanas@urv.cat

H.A. Rashwan
e-mail: hatem.rashwan@ieee.org

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_15

1 Introduction

In the last decade we have witnessed an unprecedented increase in the amount of information from citizens gathered in a variety of ways: search engines, healthcare systems, social networks, etc. In addition, video cameras have proliferated and they can be found almost everywhere: from city-scale surveillance systems controlled by local authorities, to simple private systems installed in restaurants and shops. Computerised video-surveillance databases have become a great source for data collection.

Moreover, the connection of these video-surveillance databases to the Internet allows the rapid spread of recorded videos, either because of administrators' misbehaviours or as a result of digital attacks and leaks. Regarding misbehaviour, it has been recently disclosed that managers of Aldi, a German chain of discount supermarkets, secretly filmed female shoppers and payment card readers where customers type in their PIN numbers.[1]

It could be argued that legislation should avert information leaks and misbehaviour. In fact, according to legislation [1], pictures and video recordings in which individuals can be recognised (the data handled in a VSS), must be considered personal data and, hence, they should be protected.

1.1 A Model for Privacy-Aware Video Surveillance System

In a privacy-aware video surveillance system (*cf.* Fig. 1), the images gathered by the camera are handled by a *Video Processing Module* that comprises two submodules: a *Detection Submodule* whose goal is to detect the ROIs (the regions of interest, for example faces or car plates) in a sequence of images; and a *Protection Submodule*, aimed to prevent individuals from being identified (thus, preserving their privacy). When the video stream is protected, it is stored as a database of video files in the *Information System*. It is assumed that users might have access to the system and retrieve a video file from the database but, since the ROIs are protected/encrypted, no identity information could be disclosed. Note that, only a *Trusted Manager* (TM) of the system, who has access to a *disclosure key* can fully access the ROIs of the recorded video. Last but not least, the TM might require the permission of the Law Enforcer (LE) to effectively access the full video in case of investigations. Hence, the problem of a TM arbitrarily disclosing videos is avoided.

[1] Covert Cameras at Discount Retailer: Aldi Store Managers Secretly Filmed Female Shoppers. SIEGEL Online International. April 30, 2012. http://www.spiegel.de/international/germany/aldi-spied-on-female-shoppers-with-hidden-cameras-a-830690.html.

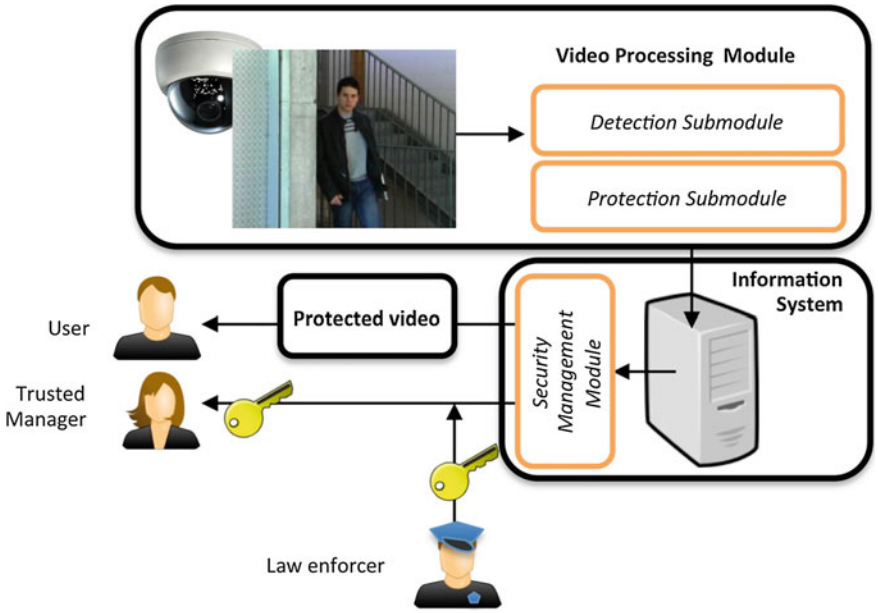


Fig. 1 Privacy-aware video surveillance system scheme

1.2 A Trustworthy Video Surveillance System

A trustworthy video surveillance system (T-VSS) is a privacy-aware video surveillance system in which two requirements are met:

1. It only stores the protected version of the video.
2. It does not require human supervision/intervention.

The first requirement is essential in order to minimise the impact of a successful attack on the Information System. Note that, if an attacker hacks the Information System, he/she will only have access to protected videos that could not be disclosed without the proper keys. Regarding the second requirement, human supervision entails an inherent lack of privacy that should be avoided. In our approach we pursuit guaranteeing the privacy of the protected video.

The techniques used in a privacy-aware video surveillance system must fulfil the next three properties in order to be trustworthy [2]:

- *Real time performance.* The procedures used in the detection and protection submodules must be executed in real time. Otherwise, some portions of the original video might be temporarily stored and a security breach could compromise the privacy of the individuals.
- *High detection accuracy.* The techniques used in the detection submodule must detect all ROIs correctly. If those techniques fail to detect them, the system will

not be able to protect the identity of some individuals and the detection process may need to be supervised by humans (which must be avoided). Note that if this module fails (even in a single video frame), privacy could be endangered for the entire video sequence.
- *Utility of the protected stream.* The techniques used in the protection submodule must protect ROIs reversibly. Hence, it must be possible to disclose the identity of the individuals appearing in a video from the protected and stored version (for instance, during a criminal investigation). This way, there is no need for storing a copy of the original video.

Despite the large amount of literature dealing with ROIs detection and protection, to the best of our knowledge, there is no proposal for a privacy-aware video surveillance system that takes into account the concepts of trust and security holistically. In this chapter we review the techniques that are used in the literature to design privacy-aware video surveillance, and propose a method for achieving trustworthy video surveillance systems.

2 Background on Current Techniques

In this section we overview the techniques in the literature according to their suitability in a T-VSS. First, we address the techniques devoted to detection of ROI, tacking into account accuracy and in real time performance. Moreover, we address the techniques for ROI protection, mainly considering the utility of the protected video.

2.1 Techniques for the Detection Submodule

We consider two trends of application in video surveillance: the face detection methods (assuming that ROIs are faces) and some more general scenarios, in which ROIs might be any moving object in the scene.

2.1.1 Face Detection

Face detection determines the locations and sizes of human faces in a scene. Its immediate application is automated people recognition. The main challenges of face detection are related to the illumination and complexity of the scene, the rotation and even the occlusion of the faces. Most of the face detection algorithms consider face detection as a feature pattern-classification problem, which uses pixels values as features. However, they are very sensitive to illumination conditions and noise. The two most common techniques for face detection are:

- *Haar-like Features* (HF). Viola and Jones [3] presented a framework for robust and rapid face detection. Haar-like features provide a good accuracy and performance

in extracting textures, and their architecture based on integral image techniques make them computationally efficient.
- *Local Binary Pattern* (LBP). Hadid et al. [4] proposed new rotation invariant and computationally lighter feature sets for face detection. Although the LBP feature is simple and can distinguish between faces and non-faces faster than HF, it suffers from environmental changes in the scene. Also, it is difficult to determine the threshold used to differentiate between faces and non-faces.

2.1.2 Motion Detection

Motion detection techniques allow the detection of moving objects in the scene. The two common techniques used for motion detection in a scene are presented next:

- *Background subtraction.* It consists of detecting foreground objects as the difference between the current frame and a static background of the scene, assuming a fixed camera. In general, statistical methods have been widely used for this purpose [5]. However, there are still many challenges in developing a good background subtraction algorithm, namely robustness against illumination changes, tackling the problem of the movement of small background objects such as leaves or rain, etc. The most relevant methods are *Mixture of Gaussians* (MoG, [6]), *Kernel Density Estimator* (KDE, [7]) and *Codebook Construction* (CC, [8]). Background subtraction techniques allow the detection of ROIs in real time, at an appropriate frame rate (e.g. around 15 frames per second in PAL quality), when executed in small form factor computers, such as Core i3 intel NUC (Next Unit of Computing). However they require the use of a fixed camera and hence cannot be used in modern VSS in which high definition cameras are able to pan and zoom over the scene.
- *Optical flow estimation.* These methods aim at estimating the spatial displacement of every image pixel between two sequential images. In particular, optical flow is an approximation of the local image motion based on local derivatives given consecutive images [9]. Among a large amount of families used for estimating flow fields, the *variational* approaches (or differential-based) yield the best performance to estimate the optical flow field. Moreover, they are the most widely used techniques [10]. The most outstanding variational techniques in optical flow are *Lucas/Kanade* (LK, [11]), *Horn/Shrunk* (HS, [12]), *Farnebäck* (FB, [13]) and *Bruhn/Weickert* (BW, [14]).

Although the protection of a face is widely accepted for privacy protection, identification could be performed based on other factors (clothes, gait analysis, etc.) Hence, VSS developers should consider the full body as the ROI, in the case of people (or a full car image instead of considering car plates).

Hence, focusing on the suitability of motion detection techniques in T-VSS, we have done some implementations of the aforementioned methods. We have tested them using the video sequences of the CAVIAR database [15].

Table 1 Evaluation of the trust offered by ROI detection methods according to their accuracy and their performance in real time

Methods	Success (%)	Real time (fps)	Suitable?
MOG-BS	76.8	16	No
KDE-BS	77.3	15.5	No
CC-BS	93.1	400	Yes
LK-OF	33.7	80	No
HS-OF	35.2	12	No
FB-OF	87.4	14.5	Yes
BW-OF	93.5	1.8	No

Table 1 shows the throughput, accuracy and suitability of the aforementioned algorithms. The accuracy of techniques can be classified [16] into *poor* for the interval [0, 60 %], *average* for the interval (60, 85 %] and *good* for the interval (85, 100 %], depending on the number of ROIs they detect accurately. The running time has been calculated on a Core i3 intel NUC. Throughput is considered to be in real time if it allows processing more than 10 fps. In addition, the trust in the tested techniques can be classified into *non-suitable* when (i) the accuracy of the technique is poor, (ii) when the accuracy of the technique is average and it works in real-time, or (iii) when the accuracy of the technique is good and it does not work in real-time, and *suitable* when the accuracy of the technique is good and it works real-time.

Algorithms CC and FB give a good accuracy and work in real-time, therefore they are highly recommended for implementing the detection submodule of a T-VSS.

2.2 Techniques for the Protection Submodule

In this section, we review the ROI protection techniques in the literature, paying attention to the utility of the protected video. We classify the proposals into two groups, depending on the domain in which ROIs are protected: first, *pixel domain* techniques, which modify the ROI in every frame, before compression of the video; second, *compression domain* techniques, which modify the data in the container of the compressed video.

2.2.1 Techniques on the Pixel Domain

There are several proposals in the literature dealing with ROI protection in the pixel domain, which can be classified into three trends:

- *Pixel transformation* They consist of replacing the value of a pixel (e.g. blurring, pixelisation and noise addition [17, 18], see Fig. 2). The implementation of such techniques is very simple but their application results in a non-invertible protected video (i.e. they are one-way operations). Some of these operations are achieved by

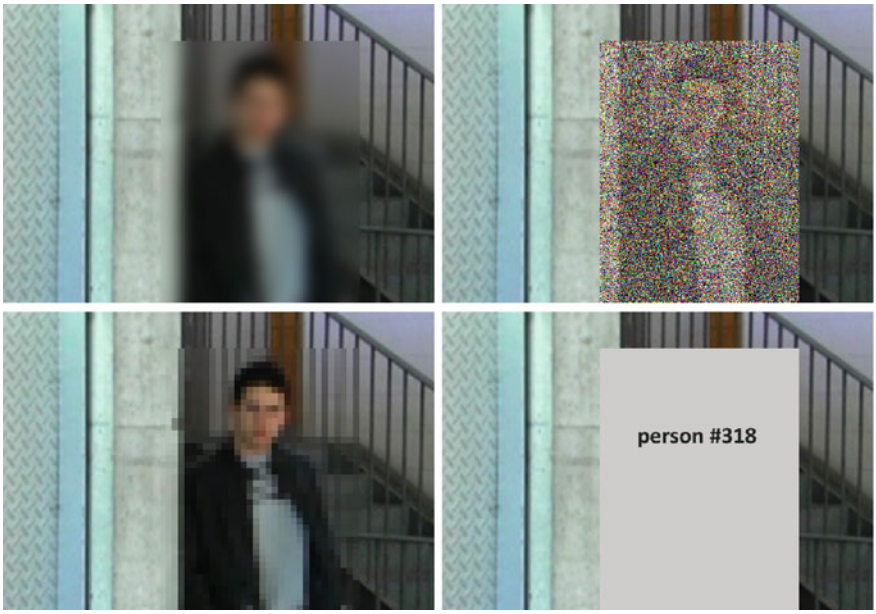


Fig. 2 Some transformations aiming at protecting ROI's privacy: blurring (*top*, *left*), noise (*top*, *right*), pixelisation (*bottom*, *left*) and information hiding with shapes (*bottom*, *right*)

means of cryptographically modifying the pixels' values [19]. If ROIs are protected using these techniques, the utility of the protected video is low: the permutation of pixels results in a set of high-frequency image blocks; then, these blocks will pass through the compression procedure, which will discard high frequency components so as to decrease the video size; as a result, protected ROIs will suffer a heavy information loss after compression and it will be difficult to obtain the original image from a compressed and protected frame.

- *Abstraction-based techniques* Those consist in replacing a ROI (e.g. a person) by a shape in the pixel domain (see Fig. 2 bottom, right). An example of those techniques can be found in [20].

All these approaches are computationally feasible. However, in all cases, the Information System must store a copy of the original video. Thus, the use of ROIs protection in the pixel domain is clearly discouraged, unless in the case of storing raw and uncompressed frames (however, note this is not a common scenario in video surveillance).

2.2.2 Techniques on the Compressed Domain

These techniques protect the ROIs during or after the image compression. For the sake of completeness, we briefly introduce the concept of compressed video:

Table 2 Summary of the protection techniques according to their suitability in a T-VSS

Domain	Method	Comment	Suitable?
Pixel domain	Transformation	Non-reversible techniques	No
	Abstraction	Need to store original video	No
Compression domain	Coefficient encryption	Full-frame encryption	No
	Sign flipping	Weak management of keys	Yes
	Information hiding	The original video is stored	No

- A compressed video is a set of compressed frames, grouped in GOPs (*Group of Pictures*). Each GOP starts with an I-frame (*intra-coded*) and contains several P-frames (*predicted*) and B-frames (*bi-predictive*).
- I-frames are stored and compressed entirely: the frame is divided into blocks; a frequency transform (e.g. Discrete Cosine Transform) is applied to each block; a quantization is applied to each block (each frequency component is divided by a number, aiming at reducing the number of discrete symbols but resulting in a lossy compression and, also, a set of zero coefficients); finally, entropy encoding (for the non-zero coefficients) and run-length encoding (for the zero coefficients) are applied for a lossless compression of the block. The information needed to reconstruct the frame is stored in a specific and standardized data structure.
- P and B-frames are not stored entirely: they merely consist of the changing blocks between frames in the GOP, described in terms of motion vectors and block differences.

If ROIs are protected in the compression domain, some values of the compressed video stream data structure might be encrypted. Hence, unauthorized users (i.e. without a proper decryption key) would obtain noise in the ROI pixel area of the decompressed frame. On the contrary, authorized users (i.e. TMs with the corresponding decryption key) would be able to decrypt the structure and hence reconstruct the original ROI. These protection techniques on the compression domain cope with the utility property. As a result, protecting ROIs in the compression domain does not require storing the original copy.

We present now the most relevant trends:

- *Coefficient encryption.* Several proposals fall into this category. In Boult [21] the authors use the DES cryptosystem to encrypt the coefficients data, but the encryption decreases the efficiency of the entropic compression of video. In

Shahid et al. [22], an AES encryption is done after the entropy encoding phase. The resulting bitrate is not modified but, unfortunately, all the frame is encrypted, without taking into account ROIs detection and protection: this does not serve the purpose of surveillance due to the fact that users could not understand the context (i.e. the scene, background, etc.) without decrypting the video.

- *Sign flipping.* There is a plethora of proposals based on scrambling the data to produce a privacy-aware video. The technique in [23] is based on flipping the sign of the coefficients of the luminance components according to a pseudo-random string. Authors use different combinations of security keys in order to produce a protected video that is robust against brute force attacks. However, this proposal does not elaborate on the secure management of the cryptographic keys involved in the process.
- *Information hiding.* In Martinez-Ponte et al. [24], the authors use the JPEG 2000 standard to protect frames, whose frames consist of a set of quality layers (each one providing more or less details depending on a quality value). Hence, the method provides trusted managers with access to all layers of the picture. On the contrary, unauthorized users would only be able to decode the lowest quality layers. Note that these VSS only store the original video and deliver specific quality layers, so the original video is in fact stored (Table 2).

As a summary, regarding protection techniques, we have seen that the cryptographic approaches tend to increase the size of the compressed video or protect the full frame. The information hiding technique based on JPEG 2000 certainly stores the original video and, thus, does not fulfill our properties for a T-VSS. Hence, in order to implement a T-VSS, techniques such as [23] could be used.

3 A Method to Implement a Trustworthy Video Surveillance System

In this section, we describe the techniques that, according to the properties defined in Sect. 1.2, can be used to deploy a T-VSS. We briefly describe the process of the Detection Submodule and focus on the Protection Submodule.

3.1 Detection Process

The process to detect the ROIs can be summarized as follows:

1. The original raw video stream is obtained from the camera controller.
2. The ROIs are detected, using either a robust optical flow technique based on tensor voting [25] or a background subtraction technique [8]. Note that using motion detection based on optical flow is essential in order to avoid the problems

of using simpler techniques under certain conditions: for instance, an outdoor camera would detect moving leaves or even rain as ROIs in case of using background subtraction. However, its implementation is more time consuming than any technique based on background subtraction.
3. An ancillary data structure containing information of the ROIs is stored.

At this point, a compressed frame is divided into 8×8-pixel blocks which are applied a frequency transform (e.g. Discrete Cosine Transform). The obtained 8×8-coefficient blocks describe the pixel block in terms of texture and details. For each block, there is one DC (a direct coefficient, with zero frequency) and 63 AC coefficients (alternate coefficients, with non-zero frequencies). Frames are grouped into successive frames, forming a GOP.

3.2 Protection Process

The ROIs protection system is constituted by the following stages applied to a given sequence of GOPs of the compressed video:

1. For each GOP, generate a seed for a pseudo-random number generator (PRNG) using the GOP number in the sequence and some other random values. Protect the seed using the secret key of the TM.
2. Protect each GOP as follows:
   - Generate the *Protection Stream PS*, a pseudo-random bit sequence of length $l = B \times (b_{DC} + 63)$, where B is the number of coefficient blocks belonging to ROIs in the compressed GOP, and b_{DC} is the number of bits for encoding the DC component of a block.
   - Protect each coefficient block b by XORing, i.e. encrypting, the i-th bit of the DC coefficient with the $(64 \cdot b + i)$-th bit of PS and flipping the sign of the j-th AC coefficient if the $(64 \cdot b + j)$-th bit of PS equals one, where b is the number of the coefficient block being protected.

This method is based on [23], but making transformations on the DC coefficient and the AC coefficients. This results in a more secure protection of the ROI without affecting the compression of the video.

In order to strengthen the storage of the TM's secret key and to involve the law enforcer, we can apply the following security measures:

- First, the seed of the PS depends on the GOP number but also on a secret value generated with a multiparty protocol by TM and the law enforcer. In a nutshell, the Protection Submodule contacts the server of the law enforcer using a secure channel and communicates the seed of the PRNG, which will be encrypted by the law enforcer server. The Protection Submodule will not store the seed value and, consequently, the participation of the law enforcer counterpart will be mandatory to disclose the video (to decrypt the stored value and to obtain the seed). Certainly,

the seed could be dishonestly used by the TM, but we assume this scenario is not possible: the TM is indeed *trusted* in this sense.
- Second, the cryptographic values and the video are stored together with MAC values in order to detect integrity failures (and make use of backup copies if necessary).

4 Implementation

In this section we present the implementation of a prototype for our proposed system. The prototype consists of a Detection Submodule and a Protection Submodule running on an Intel NUC computer. The implementation is in C language and runs on a Windows XP operating system. The Information System can be managed via a web interface. The computer is connected to an IP network via wi-fi and is equipped with a high definition webcam. Figure 3 depicts the prototype.

Regarding the protection of the privacy, we show in Fig. 4 the results for our proposal compared to the protection suggested in [23]. Our method (left) provides a better privacy protection, without increasing the size of the video file.

All processes executed in the Detection and Protection submodules must work in real time so as to reduce (to the minimum) the temporary storage of original video. Certainly, the internal components of the system use temporary buffers as a support for the software processes. Notwithstanding, we assume that the video surveillance system does not write temporary data in its file system, in this sense we understand that behaves like a "tamper-proof device". Moreover, the number of frames per second should also be considered. In this sense, with the aim to make

Fig. 3 The prototype for our proposed system

Fig. 4 The result of applying our protection scheme (*left*) and the protection described in [23] (*right*)

easy the processing in real time, the number of frames per second of the original video is kept low (e.g. 15 fps). With our prototype, and using PAL frame sizes, we could process 53.7 frames per second in case of using background subtraction with CC. When using FB optical flow, the value decreases to 11.2 frames per second. Certainly, with current small factor computers, optical flow is still far from allowing high frame rates.

Besides providing a real-time and reversible privacy preservation, the system must fulfill some security requirements. First, disclosing the identity of people appearing in the video should not be straightforward for attackers. Second, the cryptographic functions utilized in our platform must be evaluated appropriately. To cope with these security requirements, both the original video and PSs are never stored in the system. Even the seed value stored in the Information System is encrypted by the law enforcer.

5 Conclusions

In this chapter we have dealt with a comprehensive approach to privacy in video surveillance. He have not merely addressed the detection of ROIs using computer vision techniques, we have designed a whole video surveillance platform that takes into account the privacy of individuals. Firstly, we have coined the concept of Trustworthy Video Surveillance System. Secondly, we have described and analyzed the methods involved in the main steps of video surveillance systems (ROI detection and protection) that are found in the literature. We have analysed these techniques focusing on the properties that a T-VSS must fulfil (i.e. real time performance, high accuracy and utility). Finally, we have proposed and tested a combination of methods that fulfill our model of T-VSS.

Future work includes the study of the effect on the protected videos of image transformations such as rescaling and cropping. Also, we expect to implement some routines taking into account the performance and resource consumption, aiming at allowing at least 15 fps with the Intel NUC, for optical flow detection and PAL frames.

Last but not least, we recall that trustworthy video surveillance plays a key role in the so-called dataveillance society and, hence, the disciplines in Privacy Enhancing Technologies must also focus this issue.

References

1. Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 http://europa.eu/legislation_summaries/information_society/data_protection/l14012_en.htm
2. Martínez-Ballesté, A., Rashwan, H.A., Puig, D., Paniza Fullana, A.: Towards a trustworthy privacy in pervasive video surveillance systems. In: PERCOM Workshops, pp. 920–925, IEEE Computer Society (2012)
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of Computer Vision and Pattern Recognition, vol. 1, pp. 511–518 (2001)
4. Hadid, A., Pietikainen, M., Ahonen, T.: A discriminative feature space for detecting and recognizing faces. In: Proceedings of Computer Vision and Pattern Recognition, vol. II, pp. 797–804, IEEE Computer Society (2004)
5. Bouwmans, T., El Baf, F., Vachon, B.: Background modeling using mixture of Gaussians for foreground detection—a survey. In: Handbook of Pattern Recognition and Computer Vision, vol. 4, no. 2, pp. 181–199. World Scientific Publishing (2010)
6. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In Proceedings of Computer Vision and Pattern Recognition, vol. II, pp. 246–252 (1999)
7. Elgammal, A.M., Harwood, D., Davis, L.S.: Nonparametric model for background subtraction. In: Proceedings of IEEE European Conference Computer Vision, pp. 751–767 (2000)
8. Kim, K., Chalidabhongse, T.H., Harwood, D., Davis, L.S.: Background modeling and subtraction by codebook construction. In: International Conference on Image Processing, vol. V, pp. 3061–3064 (2004)
9. Weickert, J., Bruhn, A., Brox, T., Papenberg, N.: A survey on variational optic flow methods for small displacements. Math. Stat. **10**1, 103–136 (2006)
10. Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Blacket, M.J., Szeliski, R.: A Database and evaluation methodology for optical flow. Int. J. Comput. Vis. **92**(1), 1–31 (2011)
11. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the 1981 DARPA Image Understanding Workshop, pp. 121–130, April 1981
12. Horn, B.K.P., Schunck, B.G.: Determining optical flow: a retrospective. Artif. Intell. **59**(1–2), 81–87 (1993)
13. Farnebäck, G.: Fast and accurate motion estimation using orientation tensors and parametric motion models. In: International Conference on Pattern Recognition, vol. I, pp. 135–139 (2000)
14. Bruhn, A., Weickert, J., Kohlberger, T., Schnorr, C.: A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. Int. J. Comput. Vis. **70**(3), 257–277 (2006)
15. CAVIAR: Context aware vision using image-based active recognition. http://homepages.inf.ed.ac.uk/rbf/CAVIAR/
16. Chabrier, S., Emile, B., Rosenberger, C., Laurent, H.: Unsupervised performance evaluation of image segmentation. EURASIP J. Appl. Signal Process (2006) http://asp.eurasipjournals.com/content/2006/1/096306

17. Wickramasuri, J., Datt, M., Mehrotra, S., Venkatasubramanian, N.: Privacy protecting data collection in media spaces. In: Proceedings of the 12th Annual ACM International Conference on Multimedia, pp. 48–55 (2004)
18. Wactlar, H., Stevens, S., Ng, T.: Enabling personal privacy protection preferences in collaborative video observation. NSF Award, http://www.nsf.gov/awardsearch/showAward.do?awardNumber=0534625
19. Carrillo, P., Kalva, H., Magliveras, S.: Compression independent reversible encryption for privacy in video surveillance. EURASIP J. Inf. Secur. Spec Issue Enhancing Priv. Prot. Multimedia Syst. **5**, 2009 (2009)
20. Tansuriyavong, S., Hanaki, S.: Privacy protection by concealing persons in circumstantial video image. In: Proceedings of the Workshop on Perceptive User Interfaces, pp. 1–4 (2001)
21. Boult, T.E.: PICO: Privacy through invertible cryptographic obscuration. In: IEEE Workshop on Computer Vision for Interactive and Intelligent Environments, pp. 27–38 (2005)
22. Shahid, Z., Chaumnt, M., Puech, W.: Fast protection of H.264/AVC by selective encryption of CAVLC and CABAC for I and P frames. IEEE Trans. Circ. Syst. Video Technol. **21**(5), 565–576 (2011)
23. Dufaux, F., Ebrahimi, T.: Scrambling for privacy protection in video surveillance systems. IEEE Trans. Circ. Syst. Video Technol. **18**(8), 1168–1174 (2008)
24. Martinez-Ponte, I., Desurmont, X., Meessen, J., Delaigle, J.: Robust human face hiding ensuring privacy. In: Proceedings of International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), Genova, Italy (2005)
25. Rashwan, H.A., Garcia, M.A., Domenec Puig, M.A.: Variational optical flow estimation based on stick tensor voting. IEEE Trans. Image Process. **22**(7), 2589–2599 (2013)

Electronic Ticketing: Requirements and Proposals Related to Transport

M. Magdalena Payeras-Capellà, Macià Mut-Puigserver, Josep-Lluís Ferrer-Gomila, Jordi Castellà-Roca and Arnau Vives-Guasch

Abstract The use of electronic tickets (e-tickets) on mobile devices allow customers to book everywhere and use e-tickets immediately, and allows the companies to save resources and speed up management processes. Transport is one of the main sectors that use tickets in their standard activity. A wide variety of transport systems can benefit from the use of e-tickets. However, the use of e-tickets leads to various privacy abuses since anonymity of users is not always guaranteed and, therefore, users can be traced and profiles of usual movements can be created. In this chapter, we focus especially on the properties related to user privacy and we review and classify the main proposals in this area.

1 Introduction

Transport and tourism are some of the most affected sectors by the use of Information Technologies (IT). Nowadays, it is possible to easily get information of a certain destination, look for flights to that destination, book a hotel room, get museum or park tickets and so on.

M.M. Payeras-Capellà · M. Mut-Puigserver · J.-L. Ferrer-Gomila
Departament de Ciències Matemàtiques i Informàtica, Universitat de les Illes Balears,
Carretera de Valldemossa, km 7,5, 07122 Palma de Mallorca, Spain
e-mail: mpayeras@uib.es

M. Mut-Puigserver
e-mail: macia.mut@uib.es

J.-L. Ferrer-Gomila
e-mail: jlferrer@uib.es

J. Castellà-Roca (✉) · A. Vives-Guasch
Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili,
Av. Països Catalans 26, 43007 Tarragona, Spain
e-mail: jordi.castella@urv.cat

A. Vives-Guasch
e-mail: arnau.vives@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_16

Paper ticket management has costs for users and companies that could be reduced. The issuing cost of a paper ticket is low, but issuing a great amount of paper tickets has an important cost, which should be taken into account. Managing costs have to be also considered. The use of electronic tickets (e-tickets) in a company affects the business itself and also the user. Purchase and reception phases could be fully electronic, but they require the validation process to be also electronic. Users carry tickets while they are moving, and validate them in order to get access to the service. For this reason, the user must have a suitable device in order to manage and use e-tickets. Mobile devices (such mobile phones, PDAs or Smart phones) are considered the best positioned devices to these e-tickets systems. These devices offer suitable computation and storing capacities, and a rich variety in the latest wireless communication technologies (Bluetooth, NFC and also Wi-Fi). All these features are available in a reduced size, providing mobility and flexibility to these systems. In addition to the previous considerations, the real application of these e-ticket systems depends on their security, due to the ease of copy of electronic content in addition to privacy issues. E-tickets must be equal or even more secure than paper tickets.

Transport is one of the main sectors that use tickets in their standard activity. Paper tickets are progressively substituted for e-tickets, thus reducing paper costs and making all the process more dynamic. E-tickets can be used for multiple transport services. In this way, the AMSBUS [3] booking system from the Czech Republic allows the purchase of SMS tickets. First, the passenger receives the e-ticket into his mobile phone. Then, he shows the message to the ticket inspector when he is instructed to do so. In Denmark, the same kind of service is provided by Fynbus [15]. Flight companies are world leaders in the use of e-tickets and emerging IT. The International Air Transport Association (IATA) started in 2004 a programme to introduce the use of e-tickets [21], which was totally implemented in 2008. This initiative eliminates costs of ticket printers, maintenance, and ticket distribution, and represents 3 billion US dollar annual savings due to the fact that an e-ticket costs 1US dollar to process versus 10 US dollar per paper ticket [22]. Another example of that could be the electronic air flight boarding pass. Vodafone and Spanair [43] created a test, in 2007, in which passengers received their electronic boarding passes into their mobile phones. Other companies like Air Canada [1] or Continental [35] have followed the same direction and they offer similar services to their customers.

These examples prove two important facts: (i) there is a progressive introduction of e-tickets in different kinds of services; and (ii) mobile phones are the main platform for e-tickets.

We next enumerate some advantatges of the use of e-tickets on smartphones or similar devices:

- Customers are able to book everywhere, even without a printer.
- Tickets can be bought and used immediately.
- Easier and faster communication between the customers and the company takes place.
- Company saves resources and speeds up the management process.

Finally, although transport is the most representative scenario of e-ticketing use, e-tickets can also be used in other fields. The leisure sector has some examples of e-ticketing systems that are being applied. They can be used to book sport events or any other kind of live show. For instance, Leeds United [28] supporters can book a ticket for a match and later receive an SMS with the booking confirmation together with some additional information such as their assigned seats.

1.1 Electronic Ticket: Definition

The ticket is a contract between a user and a service provider. If the user demonstrates his ownership of the ticket, he obtains the right to use the service under its terms and conditions [13] (e.g. ticket validity time). Commonly, the ticket validation is required in order to use the service. Depending on the conditions of the ticket, it can be validated once, a predefined number of times or indefinitely up to a deadline.

The ticket must include elements to assure the system security and the users' privacy. The requirements related to security and privacy can vary among different applications of e-tickets. In some cases, security would be critical, such as e-ticket falsification on air travel. In others, privacy requirements, such as the anonymity of the users, are mandatory.

1.2 Objectives

The main purpose of this chapter is to construct specified knowledge in e-ticketing systems, by describing their phases and the involved participants, by defining the information stored in the ticket and by defining and describing their main security requirements and general properties in detail.

This work is going to be useful to go beyond future research based on e-ticketing systems. These systems have achieved worldwide renown and public transport can surely benefit from these technological advances due to the improvement of both security aspects presented in recent works and verification speed achieved with new portable devices.

The chapter also includes the description of the proposals related to e-ticketing developed by the authors during the research under the project ARES.

2 Electronic Tickets

This section includes the analysis of the e-ticketing systems, by first defining the involved participants, the related phases, the most suitable services related to public transport of these systems and the information to be included in the e-tickets.

2.1 Actors

We introduce the participants who are involved in an e-ticketing system, according to the authors of [12, 14, 30, 32, 34, 39]:

- User: receives the e-ticket and sends it for its validation in order to use the service.
- Issuer: issues the e-ticket to the user. E-tickets can be issued by both service providers and intermediaries [42].
- Service provider: receives the e-ticket from the user and validates it. If correct, then it provides the user access to the service.

These are the general and main participants in e-ticketing systems, but some systems include other participants. For example, a *shop* or a *broker* [12, 14, 27, 48]. Moreover, if public key cryptography systems [36, 40] are used, a *Certification Authority* (*CA*) is also included. In some cases, [41], the e-ticketing system is based on the use of Smart-Cards, so the *Smart-Card issuer* is also included in the system.

Other systems also consider the possibility to pay for the e-ticket, so that the *payment service provider*, the *bank* and the *credit card issuer* are also participants involved in the system.

2.2 Phases

According to most authors, an e-ticket system consists of three main phases: *e-ticket payment*, *issue* and *validation* [6, 9, 10, 12, 27, 30, 39, 41, 42]. However, these three phases are not unanimously defined. Some authors [32, 34, 36, 37] group payment and issue phases, converting them from three to two e-ticket phases: e-ticket issue and validation. Other proposals [4, 8, 49] add a previous registration phase, where users must be identified and authenticated in order to get permission to use the service. In [20], as well as in the previous phases, service *start* and *end* are considered. This real disagreement in e-ticket phases is due to the great number of types of services where e-tickets can be used [6, 13].

2.3 Services

The existing proposals have been evaluated depending on the services that can be offered with these systems. Since the study, one of the most relevant facts is that e-ticketing systems are mainly oriented to public transport services. Most of these transport services are rail transport [7, 10, 17–20, 24, 29, 36, 44], followed by air travel [1, 6, 7, 16, 35, 40, 43, 47, 48], bus transport [3, 7, 10, 15, 20, 29, 36, 38] and subway [7, 10, 20, 29, 36, 44], with one solely proposal used for taxi transport [7].

We can find running systems applied to tolls [7, 29–31, 44] but these are closer to electronic payment systems than e-ticketing systems. Users pay for the service when they have used it, depending on some usage factor and by charging the amount of money directly to the current or credit card accounts. This kind of services can be implemented using Automatic Fare Collection systems (AFC). A similar payment system using e-tickets is applied to Location Based Services in [2]. Also a generic e-ticket system is used in [25] as a method of service access control in a Trusted Computing environment. The rest of the proposals are not related to transport. Instead, they are oriented to the leisure sector [5–7, 26–28, 36], such as sports or cultural events.

2.4 Information

As paper tickets, e-tickets must include some basic information for their practical use. In this section, information fields that e-tickets can include are briefly described:

- Serial number (SN): unique identification of every e-ticket.
- Issuer (IS): entity who is responsible for issuing the e-ticket. This issuer can be also the service provider, or an intermediary.
- Service provider (SP): entity who offers the service to the user.
- User (US): information about the e-ticket owner. In case of existence of this field in the e-ticket, user anonymity could not be achieved.
- Service (SV): description of the service contract.
- Terms and conditions (TC): definition of the e-ticket terms and conditions, or an external link to enable consultation, alternatively.
- Type of e-ticket (TT): e-ticket includes a field describing its type.
  - Transferability (TF): if this field is permitted, transferability to another user is allowed.
  - Number of uses (NU): information about the allowed number of e-ticket uses.
- Destination (DT): this field is used for transport services in order to have user destination information.
- Attributes (AT): other attributes of the e-ticket that depend on the service (e.g. theatre seat).
- Validity time (VT): it includes two timestamps, the issue and the expiration dates.
- Date of issue (DI): e-ticket date of issue. Validity time field could be set by including this field together with the terms and conditions.
- Issuer's digital signature (DS): the e-ticket issuer has a public key cryptosystem key pair, being thus able to digitally sign the e-ticket.
- Device identification (DV): e-ticket is linked to a specific device.

3 e-Ticket Requirements

We can classify e-ticket requirements into two categories: (i) security requirements; and (ii) functional requirements. Some of them can have an impact in both categories.

We next list the most important security and functional requirements. Nevertheless, not all of the following requirements have to be met in all environments. So, scenarios will determine which requirements are more important in every case.

3.1 Security Requirements

Definition 1 (*Integrity, IT*) It has to be possible to verify whether the content of the e-ticket has been modified, as regards the one issued by the corresponding authorized issuer.

Definition 2 (*Authenticity, ATH*) A user has to be able to verify who has issued an e-ticket, and check whether the issuer is an authorized one.

Definition 3 (*Non repudiation of origin, NRO*) A user sending or generating a message has not to be able to deny it after the fact of having sent or generated it.

This requirement is particularly important because the issuer does not have to be able to deny having issued that e-ticket, and with a specific content.

Definition 4 (*Non repudiation of receipt, NRR*) A user receiving a message has not to be able to deny it after the fact of having received it.

Users that have requested and received an e-ticket have not to be able to deny having received it, as well as a provider that has received an e-ticket for a service.

Definition 5 (*Unforgeability, UNF*) Only authorized users can issue valid e-tickets.

It has not to be possible to forge e-tickets, as if they were issued by an authorized issuer.

Definition 6 (*Fairness, FR*) At the end of an exchange between two or more parties, either everybody achieves the expected items or nobody can stand in a privileged situation.

Parties are committed, in relation to a particular exchange, with fairness (everybody or nobody). This requirement can be useful for multiple processes related to e-ticket management:

- issue: if the costumer pays the amount it costs the e-ticket then she should receive a valid e-ticket from the issuer, and vice versa.
  We can think of some exceptions: donations (between users), free e-tickets (for some events), etc.

- use: if the client delivers a valid e-ticket to the service provider, the service provider must provide the service linked to the e-ticket, and vice versa.
- compensation: if the service provider has a valid e-ticket (received from a client) he must receive, if applicable, the corresponding compensation (typically economic), and if the service provider has received such compensation, then he must prove that he has received it.

A protocol for those exchanges will therefore have to be designed, and some properties achieved. A complete description of these properties can be found in [11, 33].

Definition 7 (*Non-overspending, NOV*) E-tickets can only be used as agreed between the issuer and the user.

Thus, e-tickets can be classified according to whether they can not be reused (Non-reusable see Definition 13), whether they can be used exactly a fixed number of times (Reusable see Definition 14), or they can not be used after their validity time. Period and usable times can be combined in the same e-ticket.

Overspending can be prevented if it is detected in the verification phase. Otherwise, when it is detected afterwards, it is necessary to identify the overspender.

This requirement is closely related with the uniqueness requirement of paper-based tickets. There are some techniques in order to achieve this requirement:

- tamper-resistant devices prevent a document stored in that device from being manipulable, so the distribution of these unique documents will be possible among this kind of devices. But we have to deal with the problem that these devices are not widely available.
- an entity keeps track of the used e-tickets in a centralized way, and so the uniqueness of the document is not guaranteed, but the uniqueness of the use can be guaranteed. What matters is the information on the central register.

Definition 8 (*Identified e-tickets, IDF*) Identity of the owner of the e-ticket has to be verifiable.

Not all tickets present the same requirements with regard to anonymity, so we have to imagine some possible scenarios for e-tickets: full-revocable anonymity (see Definition 9), selective-revocable anonymity (see Definition 10), and anonymous (see Definition 11).

Definition 9 (*Full-Revocable Anonymity, F-RAN*) Anonymity of users can be revoked.

Identity of users is embedded in e-tickets. Typically, only a reduced subset of actors can reveal this identity, and it will be generally done when overspending is detected during the verification process.

Definition 10 (*Selective-Revocable Anonymity, S-RAN*) Identity of a fraudulent user of an a priori anonymous e-ticket can be revealed.

This requirement is quite similar to F-RAN, but it is more restrictive, i.e. only dishonest users may lose anonymity. This requirement is better than F-RAN from a privacy point of view. However, it can require complex technical solutions.

Definition 11 (*Anonymous e-tickets, AN*) A user of an e-ticket has to stay anonymous.

Some paper tickets allow users to remain anonymous in front of the issuer, verifier and service provider. The anonymity of the user must be kept during the life cycle of the e-ticket. However, depending on the kind of payment method used, the user could be identified in this phase. Finally, the user has to be able to spend the e-ticket without any kind of identification, even if issuers and service providers collude between them.

Definition 12 (*Exculpability, EXC*) The service provider can not falsely accuse an honest user of e-ticket overspending, and the user is able to demonstrate that she has already validated the e-ticket before using it.

An honest user has to be able to prove that he has validated the e-ticket, and therefore the service provider cannot falsely accuse her.

Definition 13 (*Non-Reusability, N-REU*) The e-ticket can be used only once.

Definition 14 (*Reusability, REU*) The e-ticket can be used more than once.

In both cases (non-reusable and reusable), e-ticket overspending has to be prevented or detected.

A transport pass is an example of reusable ticket, since it can be used for several journeys (and a counter goes down in every journey) or it can be used over a period of time. E-tickets have to incorporate security measures that allow using the e-ticket in the valid period of time or for the number of uses agreed (or a combination of both, time and uses).

Definition 15 (*Transferability, TF*) One user can transfer her e-ticket to other users.

Some paper tickets can be transferred to other people (tickets for shows, bus tickets, etc.). It is not the case of identified e-tickets (plane e-tickets, etc.). If e-tickets are resold then the receiving entity has to be sure that the e-ticket is valid and not used. When we are in front of gifts or donations between confident people (a friend, familiar, etc.) no special measures have to be taken, it will be a personal matter if an overspending takes place.

However, two additional definitions of transferability must be provided: (i) weak transferability (see Definition 16); and (ii) strong transferability (see Definition 17).

Definition 16 (*Weak-Transferability, W-TF*) The e-ticket is transferable but overspending can not be verified in the transfer phase.

The e-ticket can be used by a user different from the first owner and when receiving the e-ticket the receiver will not be able to verify whether it has been provided to multiple users or if it has been used previously. The provider informs users whether the e-ticket has been previously used previously.

Definition 17 (*Strong-Transferability, S-TF*) The e-ticket is transferable and the receiver can verify that it is a valid e-ticket.

The receiver has the guarantee that only she will be able to use the e-ticket (the e-ticket has not been used). Once the ticket has been transferred, the originator does not have to be able to transfer the same e-ticket to other users.

3.2 Functional Requirements for e-Tickets

There are some other requirements that are not so directly related to security, but they can be as important as those explained previously.

Definition 18 (*Expiry date, EXD*) An e-ticket is only valid during an interval of time.

The fulfillment of this requirement can be useful in order to limit the size of database containing information of used e-tickets.

Definition 19 (*Offline verification, OFF*) E-ticket verification can be done without any external connection.

In some scenarios it will not be possible to contact external databases or Trusted Third Parties to verify whether an e-ticket is valid or not. This requirement is much related to the security mechanisms adopted.

Definition 20 (*Online verification, ON*) E-ticket verification requires a persistent connection with a trusted centralized system.

The offline option is tipically preferred, alleging costs, response time, etc.; but in an e-world where millions of transactions with credit card are made online (with "heavy" SSL connections) every day, and taking into account that there exist companies working with great computational power (Google, Facebook, etc.), it seems that this argument is no longer valid. In terms of security, online verification is better for overspending checking.

Definition 21 (*Portablity, PT*) E-tickets must be capable to be stored in mobile devices.

The use of a laptop or a personal computer to handle e-tickets is not necessary.

Definition 22 (*Reduced size, RS*) E-tickets must be as short as possible.

E-tickets are stored in mobile devices (e.g. a mobile phone, smart cards, etc..), and sometimes these devices will have a limited memory. Therefore, e-tickets have to be reduced in size as possible.

Definition 23 (*Flexibility, FX*) E-tickets can be used in multiple environments.

A specific e-ticket for each application is more expensive than adapting a general e-ticket for each application. The later solution is obviously preferred in order to economize the solution, and it will allow better security analysis.

Definition 24 (*Ease of use, EU*) Learning how to use e-tickets has to be easy.

E-tickets have to be as easy to use as paper tickets, without new problems for users.

Definition 25 (*Efficiency, EFF*) Processing an e-ticket does not have to be resource-consuming.

Mobile terminals are limited in terms of computational power and energy (battery). Thus, operations (especially communication and cryptographic operations) have to be reduced only to the necessary ones.

Definition 26 (*Payment openness, PYO*) E-tickets should be paid through usual payment systems.

When designing an e-ticket system it has to kept in mind that a payment system to obtain the e-ticket will be sometimes necessary a payment system to obtain the e-ticket. So, the e-ticket system has to allow different payment systems to be used in order to pay the e-ticket (if necessary).

Definition 27 (*Globally spendable, GS*) Costumers should be able to spend their e-tickets at any appropriate service provider.

This property is opposed to specific spendable e-tickets. In this case, e-tickets can be used only at a specific provider.

Definition 28 (*Availability, AV*) E-tickets should be usable when needed.

Denial of service attacks, disaster events or temporal malfunction of infrastructure can prevent e-ticket verification, and sometimes the event can not be delayed (a concert, a plane, etc.). A procedure to handle these situations has to be designed.

4 ARES Proposals in the Field of e-Ticketing

In the framework of the ARES project, we have several proposals related to e-ticketing applications. Our proposal ranges from a Secure Automatic Fare Collection System for Time-based or Distance-based Services with Revocable Anonymity for Users to the Tickic patent: A Secure, Anonymous and Transferable Ticketing system.

4.1 Automatic Fare Collection

Automatic Fare Collection (AFC) systems calculate the fare that users must pay depending on the time of service (time-based) or the points of entrance and exit of the system (distance-based). The progressive introduction of Information and Communication Technologies (ICT) allows the use of e-tickets, which helps to reduce costs and improve the control of the infrastructures. Nevertheless, these systems must be secure against possible fraud and they must also preserve users' privacy. In the ARES project, we have studied the security requirements for the time-based and distance-based systems and we have proposed a protocol for each of the AFC systems [23].

The protocols offer strong privacy for honest users (i.e., the service provider is not able to disclose the identity of its users and, moreover, different journeys of the same user are not linkable between them). The use of group signature schemes allows user authentication while it preserves her privacy. However, anonymity for users could be revoked if they misbehave. Our system, unlike others, does not require to obtain a new credential every time the user joins the system in order to obtain untraceability and to prevent tracking and profiling. In [23] we define new attacks based on confabulated users for distant-based AFC services. As we demonstrate in the same paper, the AFC system presented is resistant to these attacks.

Also, we have implemented the protocols in the Android mobile platform and its performance has been evaluated in two Android smartphones. The results remark that protocols are suitable to be used on the AFC system with a medium class mobile device although they offer a better experience with a high-class smartphone. The appearance in the market of more powerful mobile devices suggests a better usability of our proposal in a near future.

4.2 e-Ticketing Scheme with Exculpability and Reusability

We then presented a *Secure e-ticketing Scheme for Mobile Devices with Near Field Communication (NFC) that includes exculpability and reusability* [45]. In this context, an e-ticket is a contract, in digital format, between the user and the service provider, and reduces both economic costs and time in many services such as air travel industries or public transport. However, the security of the e-ticket has to be strongly guaranteed, as well as the privacy of their users. Accurate information about the security properties in e-ticketing schemes is given in Sect. 3 of this chapter. Our [45] e-ticketing system considers the security requirements for e-ticketing schemes and comprises three main phases: pseudonym renewal, ticket purchase and ticket verification, as well as four actors: the user, the service provider, the ticket issuer and the pseudonym manager.

We would like to highlight the exculpability property, which is a new security property that we have first introduced in the e-ticketing schemes (i.e. the service

provider can not falsely accuse the user of ticket overspending, and the user is able to demonstrate that she has already validated the ticket before using it). The system ensures that either both parties receive their desired data from other or none of them does: the protocol presents a fair-trading mechanism during the ticket verification in a way that the user pays in exchange for the right to use the agreed service (fair exchange). Another interesting property that includes our proposal is reusability. Thanks to reusability, tickets can be used a predefined number of times with the same security as single tickets. Furthermore, this scheme takes special care of the computational requirements on the user's side by using light-weight cryptography (low computational complexity cryptography and low communicational overhead.). We show that the scheme is usable in practice by means of its implementation, using mobile phones with Near Field Communication (NFC) capabilities. We analysed the global time results of our implementation for all the phases, and also the partial time results for each phase of the protocol. The results obtained, specifically the verification time (1.4 s using a 1024-bit key length in the user side, less than 1 s in the server side), allow to use the system in practice for mass-transit systems.

4.3 e-Ticketing Scheme with Transferability

E-tickets can be defined as a representation of the owners' rights to act as a user of a determined service, preserving the same requirements as the ones offered in paper format. In the same way as paper tickets, e-tickets have different properties according to the services where they are used. These services can be classified by the anonymity offered. For instance, a flight e-ticket cannot be anonymous because the identity of the passenger is a fixed parameter of it. In e-tickets with revocable anonymity, the beneficiary can use the ticket demonstrating its possession but without any need of identification. This modality helps to avoid fraud related to the reuse of e-tickets.

We created an e-ticketing system that emphasize the properties of anonymity and transferability. We have presented an e-ticketing system with anonymity and transferability based on the use of group signatures, giving a solution to enable linkability between several group signatures, and also proving their ownership with the use of Zero-Knowledge Proofs (ZKPs).

Regarding the transferability property, there are several e-ticket sales and distribution companies that allow the e-ticket transfer (http://www.ticketmaster.com/transfer or https://www.e-ticket.lu). Nonetheless, the e-ticket transfers are made through a central service and are non-anonymous. Our system transfers an e-ticket in the same way that we can transfer a paper ticket, i.e. anonymously and without the participation of a central service. We should note that we are giving the rights linked to that ticket to another user when we transfer a ticket. In some cases, it needs a change in the beneficiary role, because some service parameters are affected: the right to transfer, the service disposal and the beneficiary identity. According to the right to transfer, tickets can be granted to other users with (resale) or without any counterpart (loan).

We have proved [46] that our system has the following properties classified into two categories: security requirements, and functional requirements for e-tickets. The security requirements of our scheme are the following: (i) authenticity; (ii) non- repudiation; (iii) integrity; (iv) revocable anonymity; (v) reusability; (vi) non-overspending; and (vii) transferability. The functional requirements considered in our proposal are the following: (i) validity time; and (ii) online/offline ticket verification.

There are three main entities in the e-ticketing system, User, Issuer and Service Provider, and also a Group Manager that only interacts in case of conflict. The framework of the e-ticketing scheme has four main phases: *Ticket Issue* between the Issuer and a User; *Ticket Transfer* between two users; *Ticket Verification* between a User and the Service Provider; and finally, the *Revocation of anonymity* phase, which is only used in case of conflict, and which can be called by the Group Manager. This system guarantees the anonymity for their users and also allows the transferability of the tickets between them through payment or loan. The proposed scheme is anonymous because a group signature scheme is used. The group signature allows the issuer to verify that the user belongs to a valid group of users, but, at the same time, this issuer is not able to know the identity of the user. If the user tries to commit fraud, the group manager can revoke this anonymity.

The protocol also introduces the requirement of ticket transferability between two users using a linkable group signature scheme. With this technique, group signatures from the two users are used in order to generate a ticket transfer agreement, which could be further used as an evidence proof in case of any conflict between the parties.

Due to the innovation, security and reliability of the solution adapted by this new e-ticketing scheme (the one presented in [46]), the system has a legal protection in the form of a new patent. So now we have the exclusive right to use this technology as a solution to the problem of transferability in e-ticket protocols. The name adopted by the new e-ticketing system is ***Tickic***. Of course, the patent includes the claims which define the specific properties of the Tickic system.

Now, one of the goals of our project is to transfer this technology to the real world and to find suitable companies to implement or make use of Tickic system. We are also studying the possibility to extend the exclusive right to use the system to an international environment.

5 Summary

The use of e-tickets allows users to buy, receive and validate the ticket without any need to move to a certain place to take these steps, neither to print it. The paper cost reduction, in addition to the improved processes management (payment, issue, validation, high amount of tickets management, etc), are the main advantages for users as well as service providers. But the ticket in electronic format causes users to have to carry a device in order to save and manage these tickets.

The main e-ticket proposals have been analyzed obtaining their security requirements, their phases or processes, their involved participants in these systems, and their possible oriented services.

Information in e-tickets, based on the analyzed proposals, includes the ticket serial number, the issuer entity, the service provider, the user (in non-anonymous systems), the offered service, this service's terms and conditions, the type of ticket (its transferability and its number of valid uses), the destination (in transport services), some optional attributes (depending on the service), its validity time, the ticket's date of issue, the issuer's digital signature, and finally a device identification (if the ticket is linked only with a selected device).

There is no unanimity in terms of the number of phases of the analyzed proposals, due to the multiple services that could be offered, but there are some phases that can be considered basic: ticket payment, issue/reception, and validation. Some proposals join payment and issue/reception, or alternatively, a previous registration phase is added at the moment of the ticket reception.

The participants involved in an e-ticketing system are: the user (who obtains and validates the ticket), the ticket issuer, and the service provider (who validates the user's ticket and provides the service). Some proposals also consider the existence of intermediaries, certification authorities, etc.

Although some e-ticketing systems are used for different services, they have some common security requirements: authenticity, non-repudiation, integrity, and state. Other requirements depend on the service.

Anonymity and transferability properties are linked, that is, in order to transfer an e-ticket, this one has to be non-identified. The ticket will not be transferable if it is linked to a certain person. Anonymity cannot be achieved in case of air travel or shipping companies. In all of these cases, users have to be identified and authenticated before using the service. In rail, bus, subway and taxi companies, the ticket could be anonymous except for multitravel tickets linked to a certain person.

The number of uses is another property that could be configured, especially for mediaries, transport and leisure companies. These companies offer different services that require different modes of use. For example, a single ticket could be used only once, but multitravel tickets are used many times. Another example could be seasonal tickets, depending on the ticket validity time.

Online/offline verification depends on the availability of a communication network in the place where this verification is held. Online verification is recommendable if there is available connection to the server. For mediaries or transport companies, this property should be configured, not treated as default, because there will be places with available connection (air travel, shipping, rail and subway), places where it is being implemented (bus) and other places without it (taxi).

The great majority of systems are oriented to transport services, especially rail transport, and followed by air travel, bus and subway. Some of the toll payment system proposals use the name of e-tickets, but they are closer to a payment system than an e-ticketing system. Finally, an important note would be the incremental use of e-tickets that has been carried out in the leisure sector, especially in sports or cultural events.

Regarding the ARES project, we have briefly described the following proposals: (i) Automatic Fare Collection (AFC) system; (ii) e-ticketing scheme with exculpability and reusability; and (iii) e-ticketing scheme with transferability.

The AFC system allows two configurations: time-based or distance-based. The fare is calculated depending on the time of service (time-based) or the points of entrance and exit of the system (distance-based). Moreover, the system offers strong privacy for honest users. The service provider neither is able to disclose the identity of its users nor can it link their journeys between them. The protocols have been implemented in the Android mobile platform, showing that these protocols are suitable to be used on an AFC system with a medium class mobile device.

The exculpability property of the e-ticketing scheme (e-ticketing Scheme with Exculpability and Reusability) is a fair-trading mechanism. The user pays in exchange for the right to use the agreed service. The service provider can not falsely accuse the user, and the user is able to show that she has validated the ticket before its use. Another interesting property is ticket reusability. The computational requirements on the users' side have been taken into consideration by using light-weight cryptography. The scheme has been implemented and the results obtained allow its use for mass-transit systems.

Finally, an e-ticketing system with the properties of anonymity and transferability (***Tickic***) was proposed. The users can transfer e-tickets in the same way they do with paper tickets, i.e. anonymously and without the participation of a central service. They just need to approach their phones to transfer the ticket. If some user commits fraud, she is identified, thus revoking her anonymity. The scheme is protected by a patent application and one of our goals is to transfer this technology to the real world.

References

1. AirCanada: Mobile check-in (2007). http://www.aircanada.com/en/travelinfo/traveller/mobile/mci.html
2. Amoli, A.S., Kharrazi, M., Jalili, R.: 2ploc: preserving privacy in location-based services. In: IEEE 2nd International Conference on Social Computing/PASSAT'2010, pp. 707–712 (2010)
3. AMSBUS: (2008). http://www.svt.cz/en/amsbus/
4. Arnab, A., Hutchison, A.: Ticket based identity system for DRM. In: Proceedings Information Security South Africa. Sandton, South Africa (2006)
5. Bald, D., Benelli, G., Pozzebon, A.: The siesta project: near field communication based applications for tourism. In: IEEE 7th International Symposium on Communication Systems Networks and Digital Signal Processing (CSNDSP), pp. 721–725 (2010)
6. Bao, F.: A scheme of digital ticket for personal trusted device. In: 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC04), vol. 4, pp. 3065–3069. IEEE (2004)
7. Caron, J., Lagrange, I., Robet, L.: Contactless cell phone payment and e-ticketing: Japan leads the way at cartes and identification 2007. CARTES 2007 Press release (2007)
8. Chang, C.C., Wu, C.C., Lin, I.C.: A secure e-coupon system for mobile users. IEEE Int. J. Comput. Sci. Netw. Secur. **6**(1), 273–280 (2006)
9. Chen, Y.Y., Chen, C.L., Jan, J.K.: A mobile ticket system based on personal trusted device. Wirel. Pers. Commun. Int. J. **40**(4), 569–578 (2007)

10. Elliot, J.: The one-card trick multi-application smart card e-commerce prototypes. IET Comput. Control Eng. J. **10**(3), 121–128 (1999)
11. Ferrer-Gomilla, J.L., Onieva, J.A., Payeras-Capell, M.M., Lopez-Munóz, J.: Certified electronic mail: properties revisited. Comput. Secur. **29**(2), 167–179 (2010)
12. Fujimura, K., Kuno, H., Terada, M., Matsuyama, K., Mizuno, Y., Sekine, J.: Digital-ticket-controlled digital ticket circulation. In: 8th USENIX Security Symposium, pp. 229–240. USENIX (1999)
13. Fujimura, K., Nakajima, Y.: General-purpose digital ticket framework. In: 3rd USENIX Workshop on Electronic Commerce, pp. 177–186. USENIX (1998)
14. Fujimura, K., Nakajima, Y., Sekine, J.: Xml ticket: generalized digital ticket definition language. In: W3C XML-Dsig'99 (1999)
15. FynBus: Sms-billet (2007). http://www.fynbus.dk/
16. Granados, N., Gupta, K., Kauffman, R.: IT-enabled transparent electronic markets: the case of the air travel industry. Inf. Syst. E-Business Manage. **5**(1), 65–91 (2007)
17. Haneberg, D.: Electronic ticketing a smartcard application case-study. Master's thesis, Institut Für Informatik. Technical report 2002–16 (2002). http://www.informatik.uni-augsburg.de/lehrstuehle/swt/se/publications/2002-e_ticket_scard_app_stud/2002-e_ticket_scard_app_stud-pdf.pdf
18. Haneberg, D.: Electronic ticketing: risks in e-commerce applications. In: Digital Excellence, pp. 55–66. Springer, Heidelberg (2008)
19. Haneberg, D., Stenzel, K., Reif, W.: Electronic-onboard-ticketing: software challenges of an state-of-the-art m-commerce application. In: Pousttchi, K., Turowski, K. (eds.) Workshop Mobile Commerce. Lecture Notes in Informatics (LNI), vol. 42, pp. 103–113. Gesellschaft für Informatik (GI) (2004)
20. Heydt-Benjamin, T.S., Chae, H.J., Defend, B., Fu, K.: Privacy for public transportation. In: 6th Workshop on Privacy Enhancing Technologies (PET 2006). LNCS, vol. 4258, pp. 1–19 (2006)
21. IATA: E-ticketing (2007). http://www.iata.org/stbsupportportal/e-ticketing.htm
22. IATA: Industry bids farewell to paper ticket (2008). http://www.iata.org/pressroom/pr/2008-31-05-01.htm
23. Isern-Deyà, A., Vives-Guasch, A., Mut-Puigserver, M., Payeras-Capellà, M.M., Castellà-Roca, J.: A secure automatic fare collection system for time- based or distance-based services with revocable anonymity for users. Comput. J. (2012). doi:10.1093/comjnl/bxs033
24. Jorns, O., Jung, O., Quirchmayr, G.: A privacy enhancing service architecture for ticket-based mobile applications. In: 2nd International Conference on Availability, Reliability and Security, vol. 24, pp. 374–383, Vienna, Austria. ARES 2007—The International Dependability Conference (2007)
25. Kuntze, N., Schmidt, A.U.: Trusted ticket systems and applications. In: New Approaches for Security, Privacy and Trust in Complex Systems. IFIP International Federation for Information Processing, vol. 232 (2007)
26. Kuramitsu, K., Murakami, T., Matsuda, H., Sakamura, K.: TTP: secure acid transfer protocol for electronic ticket between personal tamper-proof devices. In: 24th Annual International Computer Software and Applications Conference (COMPSAC2000), vol. 24, pp. 87–92. Taipei, Taiwan, Oct 2000
27. Kuramitsu, K., Sakamura, K.: Electronic tickets on contactless smartcard database. In: Proceedings of the 13th International Conference on Database and Expert Systems Applications. LNCS, vol. 2453, pp. 392–402 (2002)
28. LeedsUnited: Official leeds sms (2007). http://www.leedsunited.com/page/Welcome
29. Lutgen, J.: The security infrastructure of the german core application in public transportation. In: ISSE/Secure 2007 Securing Electronic Business Processes: Highlights of the Information Security Solutions Europe/Secure 2007 Conference, pp. 411–419. Vieweg&Teubner Verlag, Vienna, Austria (2007)
30. Matsuo, S., Ogata, W.: Electronic ticket scheme for its. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. E **86A**(1), 142–150 (2003)

31. McDaniel, R.L., Haendler, F.: Advanced RF cards for fare collection. In: Commercial Applications and Dual-Use Technology, Conference Proceedings. Telesystems Conference, pp. 31–35 (1993)
32. Mihlberg, F.: On the formal analysis of e-ticketing protocols. Master's thesis, School of Computer Science and Engineering (2002)
33. Mut Puigserver, M., Payeras-Capellà, M., Ferrer-Gomila, J.L. Vives-Guasch, A., Castellà-Roca, J.: A survey of electronic ticketing applied to transport. Comput. Secur. **31**(8), 925–939 (2012)
34. Mana, A., Martínez, J., Matamoros, S., Troya, J.M.: GSM-ticket: generic secure mobile ticketing service. Gemplus World Developers Conference. Gemplus, Paris (France) (2001)
35. New York Times: Paper is out, cellphones are in (2008). http://www.nytimes.com/2008/03/18/technology/18check.html
36. Patel, B., Crowcroft, J.: Ticket based service access for the mobile user. In: Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM'97), pp. 223–233. Budapest, Hungary (1997)
37. Pedone, F.: A two-phase highly-available protocol for online validation of e-tickets. Hewlett-Packard Labs Technical Reports. HPL-2000-116 20000929 (2000)
38. P. Public. Transport: Sms tickets for public transport in prague (2007). http://www.prague.net/sms-ticket
39. Quercia, D., Hailes, S.: Motet: Mobile transactions using electronic tickets. In: 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks, Proceedings, vol. 24, pp. 374–383. Athens, Greece (2005)
40. Serban, C., Chen, Y., Zhang, W., Minsky, N.: The concept of decentralized and secure electronic marketplace. Electron. Commer. Res. **8**(1–2), 79–101 (2008)
41. Siu, I.W., Guo, Z.S.: The secure communication protocol for electronic ticket management system. In: 8th Asia-Pacific Software Engineering Conference (APSEC2001), University of Macau (2001)
42. Siu, W.I., Guo, Z.S.: Application of electronic ticket to online trading with smart card technology. In: Proceedings of the 6th INFORMS Conference on Information Systems and Technology (CIST-2001), pp. 222–239. Miami Beach, Florida (US) (2001)
43. Spanair: Spanair y vodafone españa presentan la tarjeta de embarque móvil (2007). http://www.spanair.com/web/es-es/Sobre-Spanair/Noticias-y-eventos/Spanair-y-Vodafone-Espana-presentan-la-tarjeta-de-embarque-movil/
44. Valdecasas-Vilanova, M.E.G., Endsuleit, R., Calmet, J., Bericht, I.: State of the art in electronic ticketing. Master's thesis, Institut für Algorithmen und Kognitive Systeme (2003)
45. Vives-Guasch, A., Payeras-Capellà, M.M., Mut-Puigserver, M., Castellà-Roca, J., Ferrer-Gomila, J.L.: A secure e-ticketing scheme for mobile devices with near field communication (NFC) that includes exculpability and reusability. IEICE, vol. E95-D No.1 (2012)
46. Vives-Guasch, A., Payeras-Capellà, M.M., Mut-Puigserver, M., Castellà-Roca, J., Ferrer-Gomila, J.-L.: Anonymous and transferable electronic ticketing scheme. In: Data Privacy Management (DPM), Eighth International Workshop. LNCS, vol. 8247 (2013)
47. von Dörnberg, A.:The global phenomenon of low cost carrier growth. In: Trends and Issues in Global Tourism, pp. 53–59. Springer, Berlin and Heidelberg GmbH & Co, KG (2007)
48. Wang, G., Bao, F., Zhou, J., Deng, R.H. Proxy signatures scheme with multiple original signers for wireless e-commerce applications. Vehicular Technology Conference, VTC2004-Fall, vol. 5, pp. 3249–3253. IEEE (2004)
49. Wang, S.C., Yan, K.Q., Wei, C.H.: Mobile target advertising for mobile user. International Workshop on Business and Information (BAI 2004), V2, Taipei, Taiwan (2004)

Security and Privacy Concerns About the RFID Layer of EPC Gen2 Networks

Joaquin Garcia-Alfaro, Jordi Herrera-Joancomartí and Joan Melià-Seguí

Abstract RFID systems are composed by tags (also known as electronic labels) storing an identification sequence which can be wirelessly retrieved by an interrogator, and transmitted to the network through middleware and database information systems. In the case of the EPC Gen2 technology, RFID tags are not provided with on-board batteries. They are passively powered through the radio frequency waves of the interrogators. Tags are also assumed to be of low-cost nature, meaning that they shall be available at a very reduced price (predicted for under 10 US dollar cents in the literature). The passive and low-cost nature of EPC Gen2 tags imposes several challenges in terms of power consumption and integration of defense countermeasures. Like many other pervasive technologies, EPC Gen2 might yield to security and privacy violations if not handled properly. In this chapter, we provide an in-depth presentation of the RFID layer of the EPC Gen2 standard. We also provide security and privacy threats that can affect such a layer, and survey some representative countermeasures that could be used to handle the reported threats. Some of the reported efforts were conducted within the scope of the ARES project.

J. Garcia-Alfaro (✉) · J. Herrera-Joancomartí · J. Melià-Seguí
Internet Interdisciplinary Institute, Universitat Oberta de Catalunya, Roc Boronat 117, 08018 Barcelona, Spain
e-mail: joaquin.garcia-alfaro@acm.org

J. Garcia-Alfaro
Télécom SudParis, CNRS UMR 5157 (SAMOVAR), 91011 Evry, France

J. Herrera-Joancomartí
Universitat Autonoma de Barcelona, Edifici Q, 08193 Bellaterra, Spain
e-mail: jordi.herrera@uab.cat

J. Melià-Seguí
Universitat Pompeu Fabra, Tanger 122-140, 08018 Barcelona, Spain
e-mail: joan.melia@upf.edu

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_17

1 Introduction

Radio Frequency Identification (RFID) technology is an automatic identification method for retrieving digital information without physical contact or line-of-sight, that is revolutionizing the manner in which objects and people can be identified by computers [1]. Tagging objects or even people with smart labels (the so called RFID tags) emitting identifying information in form of binary modulated signal, is the way computers can actually understand the presence of objects. RFID technology is the closest approach to the ubiquitous computing [2] or the future *Internet of Things*. RFID labels are frequently referred as the next generation barcodes. Although the utility is the same (the identification of an object), RFID offers two main advantages over conventional barcode systems. On the one hand, optical barcodes only indicates the generic product, whereas an RFID tag can identify the item (being able to distinguish different objects from the same product). On the other hand, there is no need of line-of-sight. Thus, while optical barcodes must be identified one by one, RFID tags can be read much faster, without human intervention and in large quantities [1, 3].

The unassisted wireless identification makes the RFID very attractive in areas like product traceability, inventorying or personal identification, but it also creates setbacks. Like the rest of wireless information technologies, RFID information transferred between sender and receiver is not completely secure. The air interface is much more insecure than the wired one, because the only presence of an attacker in the communication area gives him the opportunity to obtain information in a malicious way. The scarce available energy on tags, and the limited computational capabilities of tags are also determinant for security in RFID. In addition, RFID is very related with personal identification. Imagine, for instance, a medical application in wich the patient is using RFID tagged drugs. With some trivial techniques [3, 4], it will not be difficult to link patients and drugs by simply eavesdropping the exchange of messages at the RFID layer. Privacy issues must, therefore, be considered.

In this chapter, we describe those aforementioned threats and survey current countermeasures to handle them. We focus our interest on a particular RFID technology, namely the Electronic Product Code Class 1 Generation 2 (EPC Gen2) [5] standard. EPC Gen2 is a low-cost passive RFID technology for UHF, designed by EPCglobal [6] and developed in the MIT Auto-ID labs. This technology is being widespread in the retail industry [7], and also other sectors [8], thanks to the reduced price of their tags. EPC Gen2 was designed giving priority to reduce the price by means of a very simple performance [3]. Indeed, the price is the main reason for the industry to adopt or to refuse a technology. It is not a coincidence that the EPC technology appearance coincided with the explosion of RFID adoption in the retail industry [9], because tag price should not increase the product cost [3]. It can be said that a small area chip (thus a few logical gates) and no battery on-board (thus using radio frequency waves to energize the tag) will be a cheap tag. But that also means that there is almost no place for additional capabilities in the chip like security mechanisms. In fact, security measures implemented on those devices are scarce and are basically reduced to the use of pseudorandom number generators and short passwords [1].

Chapter Organization: Section 2 introduces the EPC Gen2 technology characteristics. Section 3 presents our classification of threats. Section 4 surveys recent countermeasures to handle the threats. Section 5 closes the chapter.

2 The EPC Gen2 Standard

The EPC technology is based on the use of RFID. This technology is intended to be the successor of the nowadays ubiquitous barcodes. Designed in the Massachusetts Institute of Technology Auto-ID Labs, and developed by the EPCGlobal consortium [6], the EPC technology represents the key component of an architecture known as EPCglobal Network [5]. The main components of the RFID system are the electronic labels or tags, the readers and the Information Systems (IS) e.g. middleware, databases and servers. The main goal of this architecture is the object-in-motion automatic identification in the supply chain and factory production.

The EPC Gen2 tags are passive devices powered by the electronic field generated by the reader, due to the absence of on-board batteries. A summary of their properties is provided in Table 1. EPC Gen2 tags work worldwide on the ultra high frequency (UHF) band between 860 and 960 MHz, depending on the RF regulations for each continent. The communication range between tags and readers depends on the electric field, thus, it may vary depending on the power supply and antenna design, but also on the kind of surface where the tag is placed. RFID tags are intended to be deployed widely so they must be cheap. EPC Gen2 Tags are composed by two main elements, the *Integrated Circuit* (IC) and the *antenna*.

The IC is based on a state machine model that processes and stores the RFID information. The antenna is intended to receive and transmit RFID signals, and also to energize the IC. In a low-cost RFID system, like EPC Gen2, the tags are very simple and resource limited, allowing to reduce their cost under the 10 cents of US dollar [10]. This reduction on the tag cost is proportional to the size of the silicon IC. The typical measure of space in silicon ICs is the *gate equivalent* (GE) that is equivalent to a boolean *two-input NAND gate*. The estimations on available GE for EPC Gen2 implementations are around 10,000 GE [11, 12].

Table 1 EPC Gen2 tags main properties

Identification	96 bit
Communication range	∼5 m
Tag power consumption	∼10 μW
Frequency (Europe)	865–868 MHz (UHF)
Tags Tx ratio	40–640 kbps
Tags Rx ratio	26.7–128 kbps
Identifications per second	∼200

The EPC Gen2 system communication model is common to other low-cost RFID systems where the reader (or radio-frequency interrogator) talks first. EPC Gen2 tags are passive and power dependent from the reader to respond the queries. The communication between tag and reader in the EPC Gen2 system is organized in three stages. In the *Selection* and *Inventorying* stages, the reader initiates the communication sending identification queries. The available tags in the communication range respond with a 16-bit provisional identifier extracted from the on-board pseudorandom number generator. When the reader acknowledges the provisional identifier, each single tag sends an identification sequence. The EPC Gen2 standard defines the identification sequence with 96 bits [5], but other identification sizes can be used depending on the tag manufacturer. If the reader manages to access or modify the tag memory content at this point, the *Access* stage is started. In the remainder of this section we introduce the main properties of the EPC Gen2 technology assumed in this chapter.

2.1 Tag Memory Details

An EPC Gen2 tag memory is logically divided into four banks (cf. Table 2):

- *Reserved* This memory block shall contain the 32-bit access and kill passwords. If these passwords are not specified, a logic *zero* is stored on that memory area. Tags with a *non zero* access password have to receive that value before transitioning to a secure state.
- *EPC* This block contains the Protocol Control (PC) bits and the 96-bit identification code (denoted as EPC) that identifies the tag. This memory block also contain a CRC-16 (defined in ISO/IEC 13,239) checksum of the PC and EPC codes.
- *TID* This area of memory shall contain an 8-bit ISO/IEC 15,693 class identifier. Moreover, sufficient information to identify the custom commands and optional features supported by the tag is also specified in this memory block.
- *User* This memory block is not mandatory thus, the block size is not specified in the standard. Instead, the User memory is factory-configured depending on the manufacturer.

2.2 Communication Protocol and Processes

EPC Gen2 tags do not have a power source. Instead, tags are passively powered following a very basic protocol. Tags can only respond after a message is sent by the reader. Regarding the physical layer, the reader powers up the tag by transmitting a radio frequency (RF) continuous wave to the tag, and the tag backscatters a signal to the reader using the modulation of the reflection coefficient of its antenna. RFID

Table 2 EPC Gen2 tag's memory logic map

User:	Optional
TID:	TID [15:0]
	TID [31:16]
EPC:	XPC_W1 [15:0]
	EPC [15:0]
	⋮
	EPC [95:79]
	PC [15:0]
	CRC [15:0]
Reserved:	Access password [15:0]
	Access password [31:16]
	Kill password [15:0]
	Kill password [31:16]

passive tags are powered through the electromagnetic waves received from the interrogator. Only a small fraction of the power emitted by the interrogator is received by the RFID tag antenna, inducing a voltage to the RFID tag IC. The European Telecommunications Standards Institute (ETSI) regulates the RF spectrum for the European region. It allows for the RFID UHF communication a maximum transmission power of 2 W from EPC Gen2 readers. According to the *Friis transmission equation* (cf. Eq. 1) [13], the signal power received by an RFID tag IC depends on the power signal from the reader, the gain of the antennas of both tag and reader and the inverse of the free-space path loss (FSPL) equation.

$$P_{RX,tag} = P_{TX,reader} G_{reader} G_{tag} \left(\frac{\lambda}{4\pi d} \right)^2 \tag{1}$$

The FSPL for the UHF frequency, which in Eq. 1 is represented by its wavelength (λ), decline quadratically (order of magnitude) with the distance (d) to the interrogator antenna. The communication distance d for the RFID tags depends on the factors included in Eq. 1 and it is usually considered of about 5 m, i.e., the maximum distance where the signal power is sufficient to activate the tag IC. Figure 1 shows the approximated tag received power curve depending on the distance between reader and tag. This distance is considered in ideal conditions but, on real RF environments, there are mitigation factors reducing such distance. Signal reflection, absorbing materials or inadequate antenna orientation are possible factors for reducing the communication distance. The communication is half-duplex. Simultaneous transmission and reception is not allowed.

The communication between reader and tags in the EPC Gen2 protocol is organized in reader stages and tag states. Next, the three reader stages are described (cf. Fig. 2):

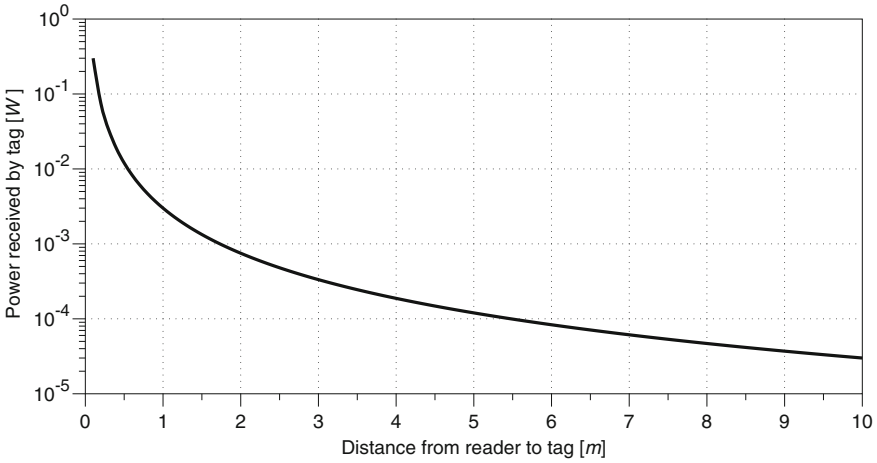


Fig. 1 At 5 m, an EPC Gen2 tag receives around 100 μW from the reader

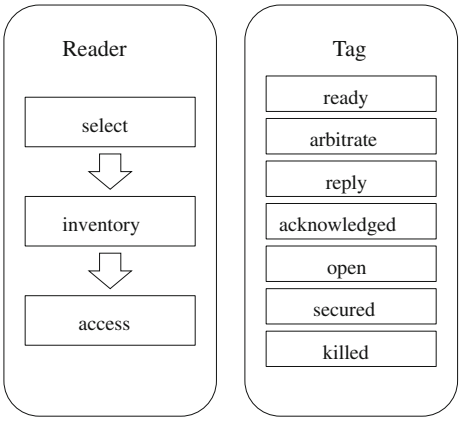


Fig. 2 Reader stages and tag states for the EPC Gen2 protocol

- *Select* In this stage, the reader selects a subset of the tag population in the communication range for inventory and access using one or more *Select* commands.
- *Inventory* The process by which a reader identifies tags. An inventory round is initialized by the reader sending *Query* commands. One or more tags may reply, thus, the tags use an anti-collision protocol to avoid collisions [5]. After *selection* the tag loads a random slot counter between *zero* and $2^Q - 1$ (with $0 \leq Q \leq 15$, automatically adjusted or user-defined) decreasing one unit for each *Query* command reception. When the counter reaches the value *zero*, the tag initiates the reply. If the reader detects a single tag reply, it requests the identification from the tag. Figure 3 shows an example of a reader inventorying a single tag.
- *Access* The process by which a reader modifies or reads individual tags' memory areas. This stage can only be initiated after a successful inventory process.

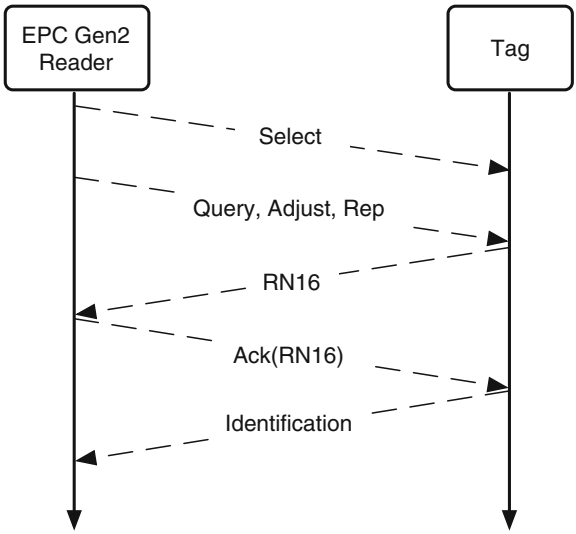


Fig. 3 Example of *Select* and *Inventory* process

The following paragraphs describe each of the possible tag states (cf. Fig. 4):

- *Ready* After being energized, a tag enters in the *ready* state. The tag shall remain in this *ready* state until it receives a *Query* command. Tag loads a Q-bit number from its pseudorandom number generator, and transitions to the *arbitrate* state if the number is *non-zero*, or to the *reply* state if the number is *zero*.
- *Arbitrate* A tag in an *arbitrate* state shall decrement its slot counter every time it receives a *QueryRep*, transitioning to the *reply* state and backscattering a 16-bit identifier (hereinafter denoted as RN16) when its slot counter reaches *zero*.
- *Reply* A tag shall backscatter a RN16, once entering in the *reply* state.
- *Acknowledged* If a tag in the *reply* state receives a valid acknowledge (*Ack*), it shall transition to the *acknowledge* state, backscattering its PC, EPC, and CRC-16. Otherwise, the tag returns to the *arbitrate* state.
- *Open* After receiving a *Req_RN* command, a tag in the *acknowledge* state whose access password is *non-zero* shall transition to the *open* state. The tag backscatters a new RN16 that both reader and tag shall use in subsequent messages. Tags in an *open* state can execute all access commands except *Lock* and may transition to any state except *acknowledge*.
- *Secured* A tag in the *acknowledge* state, and holding an access password with *zero* value, shall transition to the *secured* state, upon receiving a *Req_RN* command. The tag backscatters a new RN16 that both reader and tag shall use in future messages. A tag in the *open* state, with an access password different to the *zero* value, shall transition to a *secured* state, after receiving a valid access command. It should include the same *handle* that was previously backscattered when the tag transitioned from the *acknowledge* state to the *open* state. Tags in the *secured* state

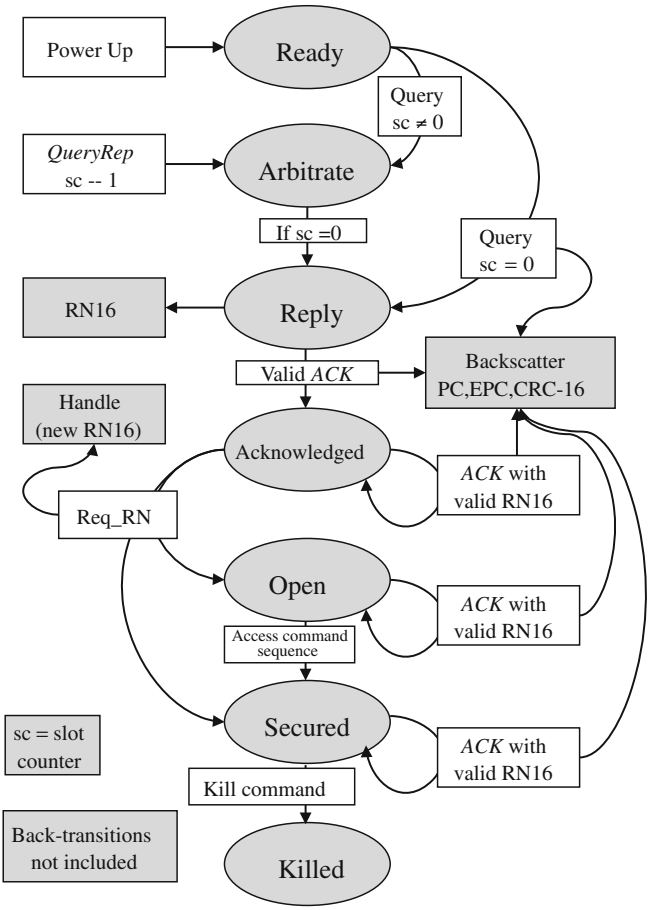


Fig. 4 EPC Gen2 tag state diagram extracted from [5]

can execute all access commands and may transition to any state except the *open* or *acknowledge* state.

- *Killed* Once a *kill* password is received by a tag in either the *open* state or the *secured* state, it shall enter the *killed* state. *Kill* permanently disables a tag. A tag shall notify the reader that the killed operation was successful, and shall not respond to any further interrogation thereafter.

3 Classification of Threats

As many other communication systems, the RFID level of the EPC Gen2 standard can be affected by threats concerning the security of the information managed by the system, and the privacy of users holding tagged objects. For this reason, it is important to determine the nature of these threats and identify the possible adversaries, to be able to analyze the security measures to adopt and under which circumstances shall be implemented. Threats targeting the security and privacy of the transmitted information in an EPC Gen2 system, are specified by the tagged object intrinsic value, or the derived value from the correlation of the tag identification with the user being identified [14].

3.1 Adversary Model and General Definitions

Prior to listing the threats, we provide some necessary definitions, such as communication parameters and expected adversary powers. We also define the abilities and goals for both parties. We start by listing the set of entities assumed in our system scenarios, and their main parameters.

- *Authorized reader* A reader registered in the system, being able to access the tag restricted memory contents. We assume that an authorized reader can read and write in the tags.
- *Legitimate tag* A tag registered in the information systems (IS), previously identified by an authorized reader.
- *Non authorized reader* A reader not registered in the IS, but having access to the EPC Gen2 communication range.
- *Illegitimate tag* Fraudulent tag accessing the EPC Gen2 system communication range. For example, a cloned tag is an illegitimate tag identification copied from a legitimate tag.

We define now some of the channel properties. We recall that in any EPC Gen2 setup, the identification tags are energized from the output power of the reader through radio-frequency waves. The communication channels are defined next, paying attention at possible security issues:

- *Reader-tag channel* Communication from reader to tag. To achieve the maximum communication distance of 10 m, transmission from reader is performed at a higher power (2–4 W) compared with the tag transmission ($\approx 10^{-4}$ W). Because of this, the reader-tag channel can be eavesdropped from hundreds of meters from the transmission point [3]. The EPC Gen2 communications protocol solves this issue giving the option to encrypt the information sent from reader to tag with a one-time-pad cover coding technique.
- *Tag-reader channel* Communication from tag to reader. Since the tag performance is powered by the reader backscattered power signal, the on-board computation

resources are scarce. In fact, the tag-reader channel is mainly used, besides the tag identification, for reader commands acknowledgment and the transmission of the pseudorandom number generated nonces used to encrypt the reader-tag communication. In this sense, the weak tag-reader channel is used to exchange the ciphering keystream between reader and tag. Hence, all the information transmitted by the tag is in plaintext.

We have seen in Sect. 2 that the EPC Gen2 standard defines three basic stages for the communication between readers and tags: select, inventory and access, and a number of possible tag states for each communication stage. Select and inventory stages are related to the tag identification process, which is the basic functionality of the system. If the tag memory content has to be modified, then the Access Stage is necessary. The two basic interaction models between tag and reader are described next.

- *Identification* To identify a tag, an EPC Gen2 reader uses two different stages. First the reader selects all the available tags in the communication vicinity in the stage known as Selection. To perform the identification of individual tags, the reader starts the Inventorying processes sending query commands to the selected tags (legitimate or illegitimate, due to the absence of authentication processes at this stage). The tags respond sequentially by using an anti-collision technique, sending its identifier in plaintext. At this point, the identification process is finished.
- *Access* Once the tag has been identified, a reader (authorized or non authorized) activates the process to access the tag memory content to read or write in it. Access queries to an EPC Gen2 tag memory are: read, lock, blockwrite, blockerase and block permalock. Access queries with the one-time-pad encryption mechanism are: write, kill and access [5].

We move now to define some of the parameters related to the adversary entities. For the EPC Gen2 system adversary model, a larger distance between tags and readers than the tag-reader communication range is assumed (unless the contrary is specified). The reason to prioritize the threats over the tag-reader channel is due to the chance of eavesdropping the information of the reader-tag channel from hundreds of meters away by using a compatible EPC Gen2 equipment. The following list of related definitions are based on [15].

- *Attack* Attempt to gain unauthorized access to a service, resource, or information; or the attempt to compromise the integrity, availability, or confidentiality. Note that success is not necessary.
- *Attacker, intruder or adversary* Originator of an attack.
- *Vulnerability* Weakness in the system security design, implementation, configuration or limitations that could be exploited.
- *Threat* Any circumstance or event (such as the existence of an attacker and vulnerabilities) with the potential to adversely impact a system through a security breach.
- *Risk* Probability that an attacker will exploit a particular vulnerability, causing harm to a system asset.

- *Passive adversary* Is the entity trying to exploit a vulnerability inside the system to execute the threat [16]. It is limited to eavesdrop information in the communication range without leaving presence evidences in the system.
- *Active adversary* Like the passive adversary, but able to transmit and receive information in the communication range. In the case of being placed in the tag-reader communication range, an active adversary is able to modify the tag memory content.

We move now to provide some basic weaknesses related to the wireless communication channel, and the lack of security measures for the information exchange between readers and tags. Although the reader-tag communication can be encrypted, the encryption keys are sent as plaintext data over the tag-reader channel. This fact leads to a vulnerability being susceptible to be attacked by an adversary.

For example, the use of pseudorandom number generators with poor statistical properties, or a certain degree of predictability, may suppose a serious risk in the communication confidentiality. A non authorized reader may access the reader-tag channel of authorized readers and legitimate tags, and analyze the generated pseudorandom sequences predictability in an Access Stage. If the adversary is able to decrypt the pseudorandom generation mechanism, a simple bitwise XOR operation between the eavesdropped and the predicted sequences will be enough to reveal the message. In that way, a non authorized reader in the reader-tag channel range may get access to the tag reserved memory areas, e.g., the kill and access passwords.

The next step in order to analyze the security of EPC Gen2 systems is to classify the main threats an adversary can take advantage. These threats are the consequence of the three basic vulnerabilities that can be pointed out when analyzing an EPC Gen2 system:

- The EPC Gen2 communication channel is weakly protected.
- Any EPC Gen2 compatible reader can obtain information from the tags in the communication channel.
- The tag design is optimized to reduce its cost. The tag capacity is very reduced and lacks of reliable authentication and security mechanisms.

The remainder of this section describes some important threats to the EPC Gen2 system security, with the corresponding vulnerability to be exploited by an adversary. The threats are grouped with regard to the targeted properties. First, we present some threats targeting confidentiality and privacy properties. Second, threats targeting integrity properties. Finally, threats targeting availability. A more detailed and methodological analysis of the threats is available in [16].

3.2 Eavesdropping, Rogue Scanning and Privacy Threats

In any passive RFID system, the reader provides a strong power signal to energize the tags. In the EPC Gen2 technology, this fact has a major relevance, since the tags

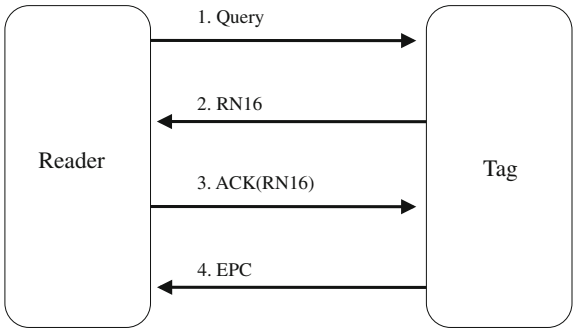


Fig. 5 Inventory protocol of an EPC Gen2 tag

may reply from larger distances. Illegitimate collection of traffic might be slightly protected by reducing the transmission power or by sheltering the area. It is, although, theoretically possible to conduct eavesdropping attacks. Two main types of eavesdropping are possible: (1) forward eavesdropping and (2) backward eavesdropping. Forward eavesdropping often refers to the passive collection of queries and commands sent from readers to tags, e.g., collection of queries and acknowledgments (cf. Steps 1 and 3) depicted in Fig. 5. Backward eavesdropping refers to the passive collection of responses sent from tags to readers, e.g., collection of control sequences and identifiers (cf. Steps 2 and 4) depicted in Fig. 5. Most authors consider that the range for backward eavesdropping could be only of a few meters [17], and probably irrelevant for a real eavesdropping attack. However, the distance at which an attacker can eavesdrop the signal of an EPC reader can be much longer. In ideal conditions, for example, readers configured to transmit at maximum output power, the signal could be received from tens of kilometers away. Analysis attacks inferring sensitive information from forward eavesdropping, for example, analysis of the pseudorandom sequences generated by the tags, are hence possible. See, for instance, results published in [18, 19], about practical eavesdropping of control data from EPC Gen2 queries with programmable toolkits, and the analysis of the obtained sequences to derive statistical artifacts of the tag components (e.g., their pseudorandom number generators).

Moreover, we have already observed in previous sections that any compatible Gen2 reader can access the EPC tags, and request their information. These operations are not properly authenticated. Therefore, it is also possible the unauthorized presence of readers in the reader-tag channel with the goal of performing fraudulent scanning of tags, i.e., performing rogue scanning attacks [17]. Although the distance at which an attacker can perform a rogue scanning is considerably shorter than the distance for eavesdropping the reader queries, the use of special hardware (e.g., highly sensitive receivers and high gain antennas) could enable rogue scanning attacks at larger distances. This clearly affects to the confidentiality of the transmitted data, which becomes highly vulnerable. Indeed, the rogue scanning threat is specially relevant because the identification code of an EPC Gen2 may reveal sensible information such as the brand, model or product cost of the tagged object. Also the production

or distribution strategies from a company can be obtained. In that way the adversary may obtain an economic benefit from selling this information for industrial espionage reasons [20].

Observe that the lack of a strong authentication process in the EPC Gen2 technology has serious consequences to the privacy of tagged object bearers. The unauthorized interrogations of EPC Gen2 tags shall give attackers unique opportunities for the collection of personal information (and without the consent of the bearer). This can also lead location tracking or surveillance of the object bearers. An attacker can distinguish any given tag by just taking into account the EPC number. Therefore, when the tags are used to identify people or wearable objects (like clothes), threats to the privacy shall be considered and properly handled [4].

3.3 Tampering, Spoofing and Counterfeiting Concerns

EPC Gen2 tags are required to be writable [21]. To protect the tags from unauthorized activation of the writing process, tags implement an on-board access control routine, based on the use of 32-bit passwords. Other integrity actions, such as the self-destruction routine of EPC Gen2 tags, are also protected by 32-bit passwords. Via the access control routine, it is possible to permanently lock or disable this harmful operation. In fact, tags are often locked by default in most of today's EPC applications, and must be unlocked by legitimate readers. Forward eavesdropping can be used by passive adversaries in order to deriving and unlocking such process [18]. Other techniques to retrieve the passwords have also been reported in the literature. For example, in [22] the authors present a mechanism to retrieve passwords by simply analyzing the radio signals sent from readers to tags. Although the proof-of-concept implementation of this technique is only available for Gen1 tags [21], the authors state that Gen2 tags are equally vulnerable.

The aforementioned attacks enabled by retrieving the passwords, that protect the writing of EPC Gen2 tags, can also be used to obtain the legitimate tag identification. This information can be reproduced on illegitimate tags, for example by means of skimming attacks [23]. If the *tag-reader* communication channel can be reached, a non authorized reader may perform active attacks like replay or scanning to obtain the information directly from the tags. Similarly, and once bypassed the password-driven routines, an EPC Gen2 authorized reader is not able to distinguish an illegitimate tag from a legitimate one. This vulnerability of the EPC Gen2 system represents a threat known as counterfeiting, since the memory of a tag can be easily modified or reproduced in the tag memory of a falsified product, what would turn into a tag cloning operation. At the same time, in a personal access system based on the EPC Gen2 technology, the identity of a person can be impersonated cloning its tag to an illegitimate one, receiving the access privileges from the impersonated person. In the context of a pharmaceutical supply chain, corrupting data in the memory of EPC tags can also be dangerous: the supply of medicines with wrong information, or delivered to the wrong patients, can lead to situations where a sick person could take the wrong drugs.

3.4 Denial of Service and Related Availability Concerns

The aim of denial of service (DoS) threats is to restrict or reduce the availability of an information system. Regarding an EPC Gen2 system, a DoS implies leaving inoperative the communication channel (either reader-to-tag or tag-to-reader channels) by making non-viable the exchange of information.

A DoS can be done in different ways. For example, taking as a reference the model introduced in Sect. 2, a radio-frequency transmitter generating noise (jamming attack) between the 865 and 868 MHz frequencies in the reader-tag channel, fills all the EPC Gen2 wireless channels avoiding authorized readers to identify the tags placed in the communication area. Even with a non-authorized reader in the reader-tag channel constantly performing identification queries, that will considerably reduce the reading efficiency of the authorized readers, delaying the system's inventorying process. In addition, the aforementioned attacks to the integrity of the tags (cf. Sect. 3.3), i.e., enabled by retrieving the tag passwords, can be used to destroy the data stored on-board of the tags, or simply to the destroy the tag itself [24]. Tag information can also be destroyed by devices that send strong electromagnetic pulses. Devices, such as the RFID-zapper [25], have been presented in the literature with such purpose. Similar effects can be obtained via de-synchronization of flawed RFID protocols [3]. Such techniques aim at misusing to the logic of the high-level protocols, rather than the on-board security primitives. Most cases show the lack of formality during the verification phase of new security techniques for low-cost RFID technologies, and can benefit from the use of formal verification [26].

4 Sample Countermeasures to Handle the Threats

EPC Gen2 security tools included in the standard [5] are basically an access password to protect certain areas of the tag memory, and pseudorandom nonces to cipher specific access commands. Additionally, low-cost RFID security related literature, brings security improvement solutions by modifying the communication protocols or the chip capabilities of the EPC Gen2 standard. In the sequel, we survey some of these solutions. First, we outline a summary of some representative research efforts conducted during the ARES project to handle those issues reported in Sect. 3. Then, we conclude with some other countermeasures proposed in the literature that we consider relevant as future directions for research.

4.1 Efforts Conducted Within the Scope of the ARES Project

During the ARES project, several improvements to the security of EPC Gen2 tag primitives and protocols were proposed. We survey some of the contribution in this

section. We classify the contributions in three main lines (lightweight authentication, security primitives improvement on tags, and secure RFID protocols), according to the types of threats they intend to address.

4.1.1 Lightweight Authentication

In a first phase, some efforts were made to handle the lack of authentication behind the eavesdropping and rogue scanning threats, while minimizing the execution of on-tag cryptographic operations. Algorithmic solutions based on secret-sharing schemes, such as those presented in [27–29] were studied and extended. The main idea is to assume that distributed secrets have been used to encrypt the EPC identifiers of a series of RFID tags. The necessary cryptographic material is split in multiple shares and distributed among multiple tags. In order to obtain the identifier of an RFID tag, a reader must collect a minimum number of shares distributed among some other RFID tags. Authentication is therefore achieved though the dispersion of secrets. The dispersion helps to improve the authentication process between readers and tags, as tags move through a supply chain. Assuming that a given number of shares is necessary for readers to obtain, e.g., the EPCs assigned to a pallet, a situation where the number of shares obtained by readers is not sufficient to reach the threshold protects the tags from unauthorized scanning (i.e., unauthorized readers that cannot obtain the sufficient number of shares cannot obtain the EPCs either). The approach can be implemented on EPC Gen2 tags without requiring any change to the current tag specification. An important problem is that privacy concerns, such as location tracking, are not addressed in the solutions reported in [27–29]. Indeed, the shares used in those approaches are static and can be misused to identify object bearers. This limitation is addressed in [30]. The extended solution relies on the use of a proactive anonymous threshold secret sharing scheme. It allows the exchange of blinded information and anonymous self-renewal of shares with secret preservation between asynchronous shareholders, with the aim of mitigating eavesdropping, rogue scanning, and tracking threats. Readers aiming at obtaining an appropriate share to unlock a tag are provided with a different new identifier per query. The solution provides the necessary guarantees to avoid linkability attacks.

4.1.2 Security Primitives Improvement on Tags

In a second phase, a series of contributions to reinforce security primitives on-board of EPC Gen2 tags were presented. Such contributions aim at addressing situations in which EPC Gen2 primitives, such as pseudorandom number generators and password-protected operations, are misused to put in place integrity and availability threats (e.g., tampering, spoofing, DoS and other similar threats). The key idea is the following. If an adversary, eavesdropping previous communications from a legitimate reader, discovers flawed generation of EPC Gen2 control sequences (i.e., pseudorandom number sequences generated by the on-board generators of the tags),

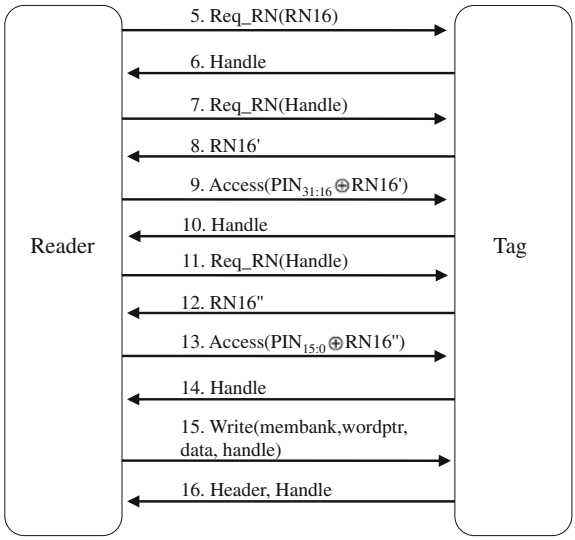


Fig. 6 Writing protocol of an EPC Gen2 tag

then he can analyze the sequences to retrieve, e.g., passwords. Assume, for instance, the protocol description depicted in Fig. 6. It presents a simplified description of the protocol steps for requesting and accessing the writing process that modifies the memory of a Gen2 tag. We assume that a select operation has been completed, in order to single out a specific tag from the population of tags. It is also assumed that an inventory query has been completed and that the reader has a valid 16-bit identifier (denoted as RN16 in Fig. 5, Steps 2 and 3) to communicate and request further operations from the tag. Using this random sequence (cf. Fig. 6, Step 5), the reader requests a new descriptor (denoted as Handle in the following steps). This descriptor is a new random sequence of 16 bits that is used by the reader and tag. Indeed, any command requested by the reader must include this random sequence as a parameter in the command. All the acknowledgments sent by the tag to the reader must also include this random sequence.

Once the reader obtains the Handle descriptor in Step 6, it acknowledges by sending it back to the tag as a parameter of its query (cf. Step 7). To request the execution of the writing process, the reader needs first to be granted access by supplying the 32-bit password that protects the writing routine. This password is actually composed of two 16-bit sequences, denoted in Fig. 6 as $\text{PIN}_{31:16}$ and $\text{PIN}_{15:0}$. To protect the communication of the password, the reader obtains in Steps 8 and 12, two random sequences of 16 bits, denoted in as RN16' and RN16". These two random sequences RN16' and RN16" are used by the reader to blind the communication of the password toward the tag. In Step 9, the reader blinds the first 16 bits of the password by applying an XOR operation (denoted by the symbol $\oplus$ in Fig. 6) with the sequence RN16'. It sends the result to the tag, which acknowledges the reception in Step 10. Similarly, the reader blinds the remaining 16 bits of the password by applying an

XOR operation with the sequence RN16", and sends the result to the tag in Step 13. The tag acknowledges the reception in Step 14 by sending a new Handle to the reader. By using the latter, the reader requests the writing operation in Step 15, which is executed and acknowledged by the tag in Step 16. Notice that an attacker can find the 32-bit password that protects the writing routine. It suffices to intercept sequences RN16' and RN16", in Steps 8 and 12, and to apply the XOR operation to the contents of Steps 9 and 13.

In [31, 32], it was reported a flawed 16-bit pseudorandom number generator design presenting the aforementioned vulnerability. The design, based on linear feedback shift registers (LFSR) for the generation of EPC Gen2 pseudorandom sequences was presented in [33, 34]. It was demonstrated that the proposal is not appropriate for security purposes, since it does not correctly handle the inherent linearity of LFSRs. A new scheme to handle the discovered vulnerability was presented in [35, 36]. The new pseudorandom number generator design, named J3Gen, still based on the use of LFSRs, relies on a multiple-polynomial tap architecture fed by a physical source of randomness. It achieves a reduced computational complexity and low-power consumption as required by the EPC Gen2 standard. It is intended for security, addressing the one-time-pad cipher unpredictability principle. J3Gen is configurable for other purposes and scenarios besides EPC Gen2 RFID technologies through two main parameters: LFSR size and number of polynomials. Its hardware complexity was studied, as well as its randomness requirements, via a statistical analysis and the power consumption through an evaluation based on CMOS parameters and SPICE language simulation.

4.1.3 Secure RFID Protocols

In a third phase, it was finally tackled the problem of flawed designs on protocols that aim at establishing some security properties on RFID environments. Security RFID protocols reported in the literature are often error-prone. A great number of protocols surveyed in [3] were reported insecure shortly after their publication. These cases show the lack of formality during the verification phase of new security techniques for low-cost RFID technologies. In [37], we deepened on this problem and illustrated how a sample protocol for the EPC Gen2 RFID technology shall be formally specified with regard to its security requirements. We defined a sample key establishment protocol, and formally verified its conformity to security properties such as authenticity and secrecy. The verification process was conducted by using the AVISPA/AVANTSSAR model checker frameworks [38, 39]. The goal was to illustrate the appropriate way of ensuring the achievement of security requirements when specifying a security protocol for the EPC technology, e.g., confidentiality properties, integrity properties, and availability properties. The proposed protocol was formally proven to achieve secure data exchange between tags and readers, based on a key generation model adapted to Gen2 RFID tags. Similar techniques could also be used to verify, as well, reader and tag primitives. Verification frameworks able to quantify weaknesses of security protocols with regard to dictionary and guessing

attacks might also help to enhance the validity of new security primitives. Some existing work in the literature on formal verification methods, such as [40–42], seem to head in this direction.

4.2 Complementary Research Directions

We conclude this section with a quick overview of complementary countermeasures that we consider relevant as future directions for research.

The first direction relies on pursuing measures based on identifier relabeling [3, 43, 44]. In a nutshell, these measures take advantage of the writable nature of EPC Gen2 tags, in order to avoid the *eavesdropping* and *spoofing* threats. Both relabeling and identifier (hereinafter denoted as ID) encryption respond to the same idea: to link in a secured database the real tag ID and a pseudo ID that can be a simple pseudonym or an encryption of the valid ID. Once the pseudonym is computed, it is written in the tag ID memory. Both pseudonym and real ID are stored in a secured database to be accessible by the system. This measure does not solve a possible counterfeiting attack to, e.g., an end-user EPC Gen2 application or any other context where tags cannot be rewritten. *DoS* is not solved by this measure, either, since tags loose their performance properties.

It could also be interesting to study physical protection of tags. Solutions such as the shielding of tags (e.g., by using a metallic bag) is proposed in [45] to avoid the activation of the tag response. Also printing on tagged objects the identifier codified in, e.g. a barcode as proposed in [46], can be understood as a backup of the legitimate identifiers, avoiding possible *spoofing* or *counterfeiting* threats, as well as *DoS*. Physical solutions could be an appropriate complement to the use of message authentication codes (MAC). The goal is to improve the integrity of the information stored in the tag. For instance, assuming a 96-bit identifier, we can use 50 bits to manage the tag ID in an EPC Gen2 application chain, and the remaining 46 bits can still be used to protect the main ID content, so to detect possible *counterfeiting* threats. The use of a *hash* function with a key k (only known by a given trusted party) can be a useful option to obtain the authentication code. This way, the final ID (96 bits) would be the result of concatenating the original ID, with the result of applying a *hash* function with key k to the *XOR* sum of k and ID_{50bits}:

$$ID_{96\text{bits}} = ID_{50\text{bits}} | H_k(ID_{50\text{bits}} \oplus k)_{46\text{bits}}$$

The operation can be done by the readers or backend servers of an EPC Gen2 application, and the result stored in the tag ID memory. Naturally, *brute force attacks* can eventually reveal the stored key. However, using an appropriate diversity of keys can improve the data integrity of most practical systems.

Some research efforts are also necessary in the field of trust, e.g., efforts with regard to trust properties of the system setups. Following the Trusted Tag Relation defined in [47], a tag is validated by an authorized party by scanning the tagged

element (e.g., by reading a tagged letter with a hand-held RFID reader connected to a back-end system). Once scanned, a status flag is marked as *valid*. The following operations in the chain of Gen2 elements would simply trust on the information provided by the scanned tag only if the step-before has been validated. This measure helps to identify more easily *counterfeiting* actions. However, it is not suitable for *eavesdropping* or *spoofing* actions because the tag is not modified in all the process. It does not handle either the *DoS* threat, since readers would probably stop working correctly. Some improvements on the Trusted Tag Relation method have been presented in [48, 49], based on a probabilistic identification protocol using collaborative readers.

5 Conclusion

EPC Gen2 systems represent one of the most pervasive technologies in the ICT field. The main feature of the EPC Gen2 technology is the tag reduced price (predicted to be under 10 US dollar cents) which means a compromise between cost and functionality. If moreover the communication between tags and readers is made in a potentially insecure channel, and that any compatible reader can access the communication between tags and readers in its communication range, the EPC Gen2 system communication has the risk of attacks on the security of the communications and the privacy of those individuals holding tagged object.

This chapter has surveyed the main characteristics of the EPC Gen2 technology and presented some of the threats and concerns reported in the related literature. It has also outlined a summary of some representative research efforts conducted during the ARES project to handle those reported threats. Particular emphasis has been made on the uniqueness of the EPC Gen2 system communications model, that only provides very basic measures for protecting the content transmitted in the reader-tag channel. The main results of this research were presented in [12, 16, 18–20, 30–32, 35–37, 50–56]. Finally, some other interesting countermeasures proposed in the literature have also been outlined. Measures such as ID relabeling or encryption can be applied in some cases due to the uniqueness of the EPC Gen2 characteristics and related applications, e.g., medical applications, to protect privacy properties.

References

1. Buttyan, L., Hubaux, J.: Security and Cooperation in Wireless Networks. Cambridge University Press (2007). http://secowinet.epfl.ch/
2. Ranasinghe, D.C., Cole, P.H.: Networked RFID systems and lightweight cryptography, chapter 3. In: Networked RFID Systems, pp. 45–58. Springer, Berlin (2008)
3. Juels, A.: RFID security and privacy: a research survey. IEEE J. Sel. Areas Commun. **24**(2), 381–394 (2006)

4. Garfinkel, S., Juels, A., Pappu, R.: RFID privacy: an overview of problems and proposed solutions. IEEE Secur. Priv. **3**(3), 34–43 (2005)
5. EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID, Specification for RFID Air Interface, Protocol for Communications at 860 MHz–960 MHz, Version 2.0.0 Ratified, EPCglobal (2013)
6. EPCglobal: The EPCglobal Website (On-line). http://www.epcglobalinc.org/. Last Access 2014
7. Motorola: RFID technology and EPC in retail, Whithe Papers (On-line). http://www.motorola.com/rfid/. Last Access 2014 (Online)
8. Potdar, M., Chang, E., Potdar, V.: Applications of RFID in pharmaceutical industry. In: IEEE International Conference on Industrial Technology (ICIT), pp. 2860–2865, Dec 2006
9. RFID Journal: Wal-Mart Opts for EPC Class 1 V2. Tech. Rep. (On-line). http://www.rfidjournal.com/article/articleprint/641/1/1/. Last Access 2014
10. Sarma, S.: Toward the 5 cents tag. Auto-ID Lab, Tech. Rep., Withe Paper Nov 2001
11. Ranasinghe, D.C., Cole, P.H.: Networked RFID systems and lightweight cryptography, chapter 3. In: Networked RFID Systems, pp. 157–167. Springer, Berlin (2008)
12. Melià-Seguí, J.: Lightweight PRNG for low-cost passive RFID security improvement. Ph.D. dissertation, Universitat Oberta de Catalunya (2011)
13. Pozar, D.: Microwave Engineering, 2nd edn. Wiley, New York (1998)
14. Avoine, G.: Adversarial model for radio frequency identification. Swiss Federal Institute of Technology (EPFL), Security and Cryptography Laboratory (LASEC), Tech. Rep. (2005)
15. Committee on National Security Systems (CNSS): National information assurance glossary. NSTISSI, Tech. Rep. 4009, May 2003
16. Garcia-Alfaro, J., Barbeau, M., Kranakis, E.: Security of self-organizing networks: MANET, WSN, WMN, VANET. In: Chapter 3, Handling Security Threats to the RFID System of EPC Networks, pp. 45–64. Auerbach Publications, Taylor & Francis Group (2010)
17. Ranasinghe, D.C.: Networked RFID systems and lightweight cryptography, chapter 18. In: Lightweight Cryptography for Low Cost RFID, pp. 311–344. Springer, Berlin (2007)
18. Garcia-Alfaro, J., Herrera-Joancomarti, J., Melia-Segui, J.: Practical Eavesdropping of Control Data From EPC Gen2 Queries With a Programmable RFID Toolkit. Hakin9, vol. 6, no. 9, pp. 14–19, Sept 2011
19. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: On the similarity of commercial EPC Gen2 pseudorandom number generators. Trans. Emerg. Telecommun. Technol. **25**(2), 151–154 (2014)
20. Garcia-Alfaro, J., Barbeau, M., Kranakis, E.: Analysis of threats to the security of EPC networks. In: Sixth Annual Communication Networks and Services Research (CNSR) Conference, Hlifax, Nova Scotia, Canada, May 2008
21. EPCglobal: The EPCglobal architecture framework. Tech. Rep. (2007). http://www.epcglobalinc.org/standards/ (Online)
22. Oren, Y.: Remote power analysis of RFID tags. Cryptology ePrint Archive, Report 2007/330, IACR (2007)
23. Hancke, G.P.: Practical eavesdropping and skimming attacks on high-frequency rfid tokens. J. Comput. Secur. **19**(2), 259–288 (2011)
24. Han, D., Takagi, T., Kim, H., Chung, K.: New security problem in RFID systems tag killing. In: Computational Science and its Applications (ICCSA, 2006). Lecture Notes in Computer Science, vol. 3982, pp. 375–384. Springer, Berlin (2006)
25. Collins, J.: RFID-Zapper shoots to kill. RFID J. (2006). http://www.rfidjournal.com/articles/view?2098. Last Access 2014 (On-line)
26. Keller, R.M.: Formal verification of parallel programs. Commun. ACM **19**(7), 371–384 (1976)
27. Langheinrich, M., Marti, R.: Practical minimalist cryptography for RFID privacy. IEEE Syst. J. **1**(2), 115–128 (2007)
28. Langheinrich, M., Marti, R.: RFID privacy using spatially distributed shared secrets. In: Ubiquitous Computing Systems, pp. 1–16. Springer, Berlin (2007)

29. Juels, A., Pappu, R., Parno, B.: Unidirectional key distribution across time and space with applications to rfid security. In: SS'08: Proceedings of the 17th Conference on Security Symposium, pp. 75–90. USENIX Association, Berkeley, CA, USA (2008)
30. Garcia-Alfaro, J., Barbeau, M., Kranakis, E.: Proactive threshold cryptosystem for EPC tags. Ad Hoc Sens. Wireless Netw. **12**(3–4), 187–208 (2011)
31. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: Analysis and improvement of a pseudorandom number generator for EPC Gen2 tags. In: Sion, R. et al. (eds.) Financial Cryptography and Data Security. Lecture Notes in Computer Science, vol. 6054, pp. 34–46. Springer, Berlin (2010)
32. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomartí, J.: A practical implementation attack on wak pseudorandom number generator designs for EPC Gen2 tags. Wireless Pers. Commun. **59**, 27–42 (2011). doi:10.1007/s11277-010-0187-1
33. Che, W., Deng, H., Tan, X., Wang, J.: Networked RFID systems and lightweight cryptography, chapter 16. In: A Random Number Generator for Application in RFID Tags, pp. 279–287. Springer, Berlin (2008)
34. Chen, W., Che, W., Yan, N., Tan, X., Min, H.: Ultra-low power truly random number generator for RFID tag. Wireless Pers. Commun. **59**(1), 85–94 (2011). doi:10.1007/s11277-010-0191-5
35. Melià-Seguí, J., Garcia-Alfaro, J., Herrera-Joancomartí, J.: Multiple-polynomial LFSR based pseudorandom number generator for EPC Gen2 RFID tags. In: IECON—37th Annual Conference on IEEE Industrial Electronics Society, pp. 3820–3825, Nov 2011
36. Melià-Seguí, J., Garcia-Alfaro, J., Herrera-Joancomartí, J.: J3Gen: a PRNG for low-cost passive RFID. Sensors **13**(3), 3816–383 (2013). doi:10.3390/s130303816
37. Tounsi, W., Cuppens-Boulahia, N., Garcia-Alfaro, J., Chevalier, Y., Cuppens, F.: KEDGEN2: a key establishment and derivation protocol for EPC Gen2 RFID systems. J. Netw. Comput. Appl. **39**(1), 152–166 (2014)
38. Armando, A., Basin, D., Boichut, Y., Chevalier, Y., Compagna, L., Cuéllar, J., Drielsma, P., Heám, P., Kouchnarenko, O., Mantovani, J., Mödersheim, S., Oheimb, O.V., Rusinowitch, M., Santiago, J., Turuani, M., Vigano, L., Vigneron, L.: The AVISPA tool for the automated validation of internet security protocols and applications. In: 17th International Conference on Computer Aided Verification (CAV'05), pp. 135–165, Springer (2005)
39. Armando, A., Arsac, W., Avanesov, T., Barletta, M., Calvi, A., Cappai, A., Carbone, R., Chevalier, Y., Compagna, L., Cuellar, J., Erzse, G., Frau, S., Minea, M., Mödersheim, S., Oheimb, D., Pellegrino, G., Ponta, S., Rocchetto, M., Rusinowitch, M., Dashti, M.T., Turuani, M., Vigano, L.: The AVANTSSAR platform for the automated validation of trust and security of service-oriented architectures. In: 18th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2012), pp. 267–282, Springer (2012)
40. Delaune, S.: Intruder deduction problem in presence of guessing attacks. In: Proceedings of the Workshop on Security Protocols Verification (SPV'03), Marseille, France, 2003, pp. 26–30
41. Groza, B., Minea, M.: A calculus to detect guessing attacks. In: Information Security, pp. 59–67. Springer, Berlin (2009)
42. Groza, B., Minea, M.: Formal modelling and automatic detection of resource exhaustion attacks. In: 6th ACM Symposium on Information, Computer and Communications Security (ASIACCS 2011). ACM, 2011, pp. 326–333
43. Wong, H., Hui, C., Chan, C.: Cryptography and authentication on RFID passive tags for apparel products. Comput. Ind. **57**(4), 342–349 (2006)
44. Weis, S., Sarma, S., Engels, D.: RFID systems and security and privacy implications. In: Cryptographic Hardware and Embedded Systems—CHES. LNCS, vol. 2523, pp. 454–469. Springer, Berlin (2002)
45. Peris-Lopez, P., Hernandez-Castro, J., Estevez-Tapiador, J., Ribagorda, A.: RFID systems: a survey on security threats and proposed solutions. In: 11th IFIP International Conference on Personal Wireless Communications. LNCS, vol. 4217, pp. 159–170. Springer (2006)
46. Juels, A., Pappu, R.: Squealing euros: privacy protection in RFID-enabled banknotes. In: Wright, R.N. (ed.) Financial Cryptography—FC'03. Lecture Notes in Computer Science, vol. 2742, pp. 103–121. IFCA. Le Gosier, Guadeloupe, French West Indies. Springer, January 2003

47. Solanas, A., Domingo-Ferrer, J., Martínez-Ballesté, A., Daza, V.: A distributed architecture for scalable private RFID tag identification. Comput. Netw. **51**(9), 2268–2279 (2007) (Elsevier)
48. Trujillo-Rasua, R., Solanas, A.: Efficient probabilistic communication protocol for the private identification of RFID tags by means of collaborative readers. Comput. Netw. **55**(15), 3211–3223 (2011)
49. Trujillo-Rasua, R., Solanas, A., Pérez-Martínez, P.A., Domingo-Ferrer, J.: Predictive protocol for the scalable identification of RFID tags through collaborative readers. Comput. Ind. **63**(6), 557–573 (2012). Special Issue on Secure Collaboration in Design and Supply Chain Management
50. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: Análisis de Seguridad y Privacidad para Sistemas EPC-RFID en el Sector Postal. In: XI Reunión Española sobre Criptología y Seguridad de la Información. Universidad de Salamanca, Salamanca—Spain, Sept 2008
51. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: Clasificación de las Amenazas a la Seguridad en Sistemas RFID-EPC Gen2. In: XII Reunión Española sobre Criptología y Seguridad de la Información, Tarragona—Spain. Universitat de Tarragona, Sept 2010
52. Melia-Segui, J., Herrera-Joancomarti, J., Garcia-Alfaro, J.: Security and privacy of postal RFID systems. In: RFIDSec Asia, Taipei, Taiwan (ROC), Jan 2009
53. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: Clasificación de las Amenazas a la Seguridad en Sistemas RFID - EPC Gen2. In: XII Reunión Española sobre Criptología y Seguridad de la Información. Universitat Rovira i Virgili, Tarragona—Spain, Sept 2010
54. Melia-Segui, J., Garcia-Alfaro, J., Herrera-Joancomarti, J.: RFID EPC-Gen2 for postal applications: a security and privacy survey. In: IEEE International Conference on RFID-Technology and Applications (RFID-TA) Guangzhou—China, pp. 118–123. IEEE, June 2010. doi:10.1109/RFID-TA.2010.5529872
55. Garcia-Alfaro, J., Herrera-Joancomarti, J., Melia-Segui, J.: A multiple-polynomial LFSR based pseudorandom number generator design for EPC Gen2 systems. In: MITACS Workshop on Network Security & Cryptography, Toronto (Canada), June 2010
56. Garcia-Alfaro, J., Barbeau, M., Kranakis, E.: Les composants RFID, sont-ils vulnerables? Techniques de l'ingenieur, no. 4–5 (2009)

Privacy on Mobile Coupons Booklets

M. Francisca Hinarejos, Andreu Pere Isern-Deyà and Josep-Lluís Ferrer-Gomila

Abstract Electronic coupons booklets are the equivalent of paper-based coupons booklets, offered to customers as a great opportunity to obtain a better offer from merchants. In this book chapter, the authors describe the main coupons booklet scenarios and identify the basic and additional security requirements. They review the state-of-the-art of the coupons booklet solutions and discuss about the main challenges: security, privacy and efficiency. In order to solve all these challenges, they present a coupons booklet scheme for the mobile scenario. They analyze their proposal to prove it meets all security and privacy requirements, and provide some performance results to prove it is a viable solution.

1 Introduction

Mobile commerce (m-commerce) represents an important area of business with a huge potential revenue for merchants and great opportunities for customers to achieve a better offer. However, one of the main concerns that negatively affects the growth of m-commerce is the lack of privacy and trust perceived by customers on merchants and on-line transactions. This is because customers want to maintain the same degree of privacy in the electronic version as in the paper version, and not always they are confident with this fact. One of the topics within the m-commerce field that suffers from this lack of privacy and trust are the mobile coupons. The true reality is that even though this topic has attracted a remarkable attention during recent years in both commercial (Gourmet, Bancotel, Groupon, LetsBonus, etc.) and scientific fields, there are no solutions covering all customer expectations (from privacy to

M.F. Hinarejos (✉) · A.P. Isern-Deyà · J.-L. Ferrer-Gomila
Department of Mathematics and Computer Science,
University of the Balearic Islands, Palma de Mallorca, Spain
e-mail: xisca.hinarejos@uib.es

A.P. Isern-Deyà
e-mail: andreupere.isern@uib.es

J.-L. Ferrer-Gomila
e-mail: jlferrer@uib.es

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_18

efficiency issues). These facts can be also extended to the field of coupons booklets, where merchants prefer selling to customers a set of coupons that are handled as a single unit (booklet of coupons). This way, merchants can establish a long-term relationship with customers, and in turn, customers can obtain a better offer.

In this book chapter we review the existing proposals for mobile coupons booklets and we make a classification of them regarding the application scenarios, their functionalities and the provided security. On one hand, we realize that a lot of those previous proposals are focused on limited scenarios in which coupons are only spendable with a single merchant. It means that customers have to maintain a relationship with every merchant with whom they want to operate and issue different coupons for each of them. In fact, this restriction is a serious drawback which slows down the growth on the use of mobile coupons. On the other hand, some of these proposals present both security and privacy problems that need to be addressed. Moreover, the vast majority of the reviewed solutions are focused on their theoretical definition, overlooking the performance evaluation and the viability study of the solution.

With the aim of providing a better solution than those previously given, we present a mobile coupons booklet scheme for the multi-merchant scenario. This solution preserves security properties from previous single merchant schemes, but in addition enhances the security, privacy and efficiency of the few solutions that deal with the multi-merchant scenario. To prove that this scheme is viable to be used by m-commerce customers, we provide some performance results.

2 Coupons Booklet

In this section we review the two general coupons booklet scenarios present in the literature, and we define the functionalities of the involved entities and the security and privacy requirements that must be accomplished.

2.1 Scenario

Reviewing previous proposals in the field of coupons booklets, we detect that three main entities are considered: customers, merchants and issuers. These entities can be defined as follows:

- *Customers*. Persons who visit merchants' shops (either online or physical), buy products, and collect or buy coupons booklets. A customer is motivated to use a coupons booklet to obtain a discount on some products or services.
- *Merchants*. Owners of shops offering products or services to customers. A merchant might distribute coupons booklets to attract customers to his shop. The customers buy the products or services at their online or physical store asking for

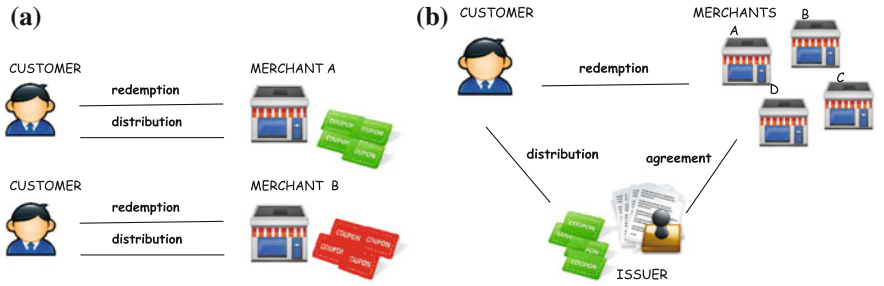


Fig. 1 Coupons booklet scenarios, **a** Single-merchant scenario, **b** Multi-merchant scenario

the offer reflected on their coupons. Thus, merchants obtain larger profits if more customers prefer their services instead of those from their competitors.

- *Issuers*. Parties in charge of the distribution of coupons booklets among a perhaps targeted set of customers. Merchants arrive at an agreement with the issuer in order to get profit of the potential customers from coupons booklets buyers.

Then, depending on the relationships among the involved entities, we identify the two following scenarios (see Fig. 1):

- *Single-merchant*. In this scenario, a merchant is in charge of the issuance, distribution and validation of coupons booklets for the services that he offers to customers. Therefore, a customer can only use coupons at the merchant who issued them [1–6] and customer has to issue a new coupons booklet for every merchant with which he wants to operate. In this case, the issuer and the merchant are the same entity, so security is controlled by the merchant because he is responsible for issuing and validating coupons. Therefore, he has all the information about the issued and redeemed coupons.
- *Multi-merchant*. In contrast to the single-merchant scenario, a customer can use the same coupons booklet at different merchants [7]. In this context, a merchant can accept coupons that were issued by another entity (another merchant or an issuer). Therefore, the security of the scheme must take into account that the merchant validating the coupons booklet must have enough and valid information to verify the coupons presented by customers. This scenario allows attracting more customers to merchants. Therefore, customers and merchants are willing to obtain greater benefits.

The latter scenario seems the most interesting one for merchants and customers, thus, in this book chapter, we will focus on this type of scenario. As explained, in this scenario the issuer is the only entity in charge of the coupons booklets issuance to customers, while merchants are the entities who accept and verify these booklets. Moreover, a merchant could accept coupons booklets issued by different issuers, but merchants should have an agreement with those issuers prior their acceptance. In our scenario, a new entity called group manager must be introduced to provide customer anonymity and anonymity revocation in case of misbehavior, as we will explain later.

2.2 Security Considerations

In the scenario considered above, some security issues must be taken into account. Next, we describe the basic security requirements that, as pointed out in [1–3], a coupons booklet solution should accomplish:

- *Unforgeability*. There is an intrinsic monetary value associated to any coupon, either explicitly or implicitly. Therefore, merchants want their coupons booklets to be unforgeable, in the sense that no coalition of customers should be able to redeem more coupons than they have been rightfully allowed, to issue new coupons booklets or to modify its content.
- *Coupon reuse avoidance*. In addition to the detection of fake coupons, the use of copies of a valid coupon must be both detected and avoided.
- *Unlinkability*. It must be unfeasible for a merchant to link a redeem procedure for a customer to the corresponding issue procedure, or to link two different redeem procedures from the same customer.
- *Unsplittability*. There are many definitions about unsplittability. On one hand, weak unsplittability, as defined in [4], requires that if a customer wants to share some coupons of a booklet with another entity, she must provide all secret information related to that coupons booklet. Thus, it is designed to discourage sharing. On the other hand, strong unsplittability requires that if a customer gives a single coupon to another customer, the first customer cannot use any other coupon until the second customer provides some information to the first one. This makes that customers sharing coupons must trust each other [7]. The property of unsplittability is not always required and it depends on the type of service offered [3].

In addition to the basic security requirements, *privacy* becomes more relevant today due to the fact customers are more and more aware about the collection and utilization of their personal data by merchants. Then, privacy deals with which information a customer reveals about herself and how to control who can access that information. Besides, the action of a merchant leaving the scenario cannot cause a security nor privacy flaw to the other parties, specially to customers and their private data.

Next, we describe the requirements associated with privacy in the considered scenario:

- *Anonymity of customers*. The confidentiality of the customer identity should be preserved, that is, customers should be able to obtain and redeem coupons without revealing any information about their identities. Neither the issuer who issues coupons to customers nor merchants that accept the coupons from customers must be able to obtain information concerning the identity of the customers.
- *Revocation of customer anonymity*. Although customers anonymity is desired, the scheme should provide mechanisms to reveal the identity of customers when they misbehave, e.g., when a customer tries to forge a coupon or to use the same coupon more than once (coupon reuse).

- *Untraceability*. Together with anonymity, privacy is also related to the impossibility to track different operations that a customer performs at different merchants or at the same merchant.
- *Confidentiality of exchanged data*. Data exchanged between participating entities (customers, merchants and issuer) must be accessible only by both edges of the communication.
- *Disaffiliation of merchants*. When a merchant leaves the system, i.e., he leaves the affiliation from the issuer, the security of the coupons booklet solution must not be compromised as well as the privacy of those information related to the customers. So, the issuer and the merchants cannot share sensitive information about customers.

3 Actual Solutions for Coupons Booklets

As we already pointed out in the introduction, there are several solutions trying to provide security for the use of booklets of coupons. In this section we analyze these solutions considering two main aspects: functionalities and security properties. Table 1 summarizes the set of features and properties of all the reviewed solutions.

Regarding functionalities, all the analyzed solutions in single-merchant scenarios [1–6], provide the basic protocols required for operating with coupons booklets (issue and redeem). However, further procedures are needed for more general scenarios, such as claim and refund. On one hand, the claim protocol is required when the entity who issues coupons and the entity who exchanges them with the customer for goods or services are not the same entity. This is the case for a multi-merchant scenario like the proposed in [7, 8]. On the other hand, a refund protocol allows customers to recover from the issuer the value of a list of already issued coupons but not used yet.

Even though the redeem process is supported by all the proposals, solutions for coupons booklets should provide mechanisms to allow redeeming more than one single-coupon within the same redeem process. This process is called multi-redemption and it is an interesting process for flexibility and efficiency purposes. However, the vast majority of analyzed solutions require executing the redeem process as many times as individual coupons are provided. To the best of our knowledge, this process is only supported by [3, 9].

As mentioned in [1, 9] a privacy protecting coupon system should at least provide the property of *weak unsplittability*. The solutions presented in [1, 2] allow *weak unsplittability* while the schemes in [4, 5, 7] obtain *strong unsplittability*. Instead, the proposal in [3] provides customer with the possibility to detach a coupon from the booklet and transfer it to another customer, but in this case, the issuer entity must be involved in the transfer process.

Concerning security, coupons booklets solutions should consider the customer privacy, and the detection and prevention of fraudulent usage of coupons. Almost all the analyzed schemes accomplish these requirements [1–5, 7], but although these schemes deal with customer anonymity, they do not take into account the

Table 1 Actual solutions for coupons booklets—a comparative analysis

	[1]	[2]	[3]	[5]	[6]	[7]	[8]	Our proposal
Scenario								
Merchant-customer relationship	SM	SM	SM	SM	SM	MM	MM	MM
Basic functionalities								
Issue	✓	✓	✓	✓	✓	✓	✓	✓
Redeem	Single	Single	Multi	Single	Single	Single	Multi	Multi
Claim	–	–	–	–	–	✓	✓	✓
Additional functionalities								
Refund	–	–	–	–	–	–	Optional	Optional
Basic security requirements								
Unforgeability	✓	✓	✓	✓	✓	✓	✓	✓
Coupon reuse avoidance	✓	✓	✓	✓	✓	✓	✓	✓
Unlinkability	✓	✓	✓	✓	–	✓	✓	✓
Unsplittability	Weak	Weak	–	Strong	–	Strong	–	Weak
Privacy requirements								
Customer anonymity	✓	✓	✓	✓	–	✓	✓	✓
Revocation of customer's anonymity	–	–	–	–	–	–	–	✓
Untraceability	✓	✓	✓	✓	✓	✓	✓	✓
Confidentiality	–	–	–	–	✓	–	✓	✓
Merchant disaffiliation	–	–	–	–	–	–	–	✓

✓YES , –NO
SM single-merchant; *MM* multi-merchant

possibility to revoke the anonymity of the customer when she makes a fraudulent use of coupons, or to provide confidentiality to the data exchange between both edges of the communication. However, the main drawback of the vast majority of coupons booklets schemes is that they are designed for *single-merchant* scenarios [1–5], so security is controlled by the merchant, since he is responsible for issuing and validating coupons. But, this type of schemes reduces the use scope of coupons booklets because customers can only interact with the merchant who issued the coupons booklets.

The authors in [6] went a step further and presented a platform where customers and merchants can be registered to use the platform functionalities; customers can obtain coupons and merchants can publish their offers. So, customers benefit from discounts offered by merchants, and merchants can obtain a great number of potential clients as well as customer profiles. However, this solution cannot be considered as a multi-merchant scenario because each merchant only offers and accepts its own coupons, but not coupons from other merchants. Moreover, the proposal has a main shortcoming, the lack of information about the provided security. Although

the confidentiality of communications is guaranteed by using SSL (Secure Sockets Layer), how coupons integrity and authentication is achieved, is not explained. In fact, the privacy of customers is not guaranteed.

The limitations of previous solutions was partially solved by Armknecht et al. [7] allowing a merchant federation and hence a multi-merchant scenario. In that scheme, customers obtain a coupons booklet from any merchant within the same federation and customers can spend the coupons at any federated merchant. The federation is an association of merchants where all merchants share a key pair (public and private), and each merchant has a different private key to sign coupons. However, the merchant who receives a coupon, previously issued by another merchant, must find and contact the original issuer in order to recover the applied discount. Moreover, merchants must share the same federation private key and a common database where data about issued and already used coupons must be updated by merchants. So, when a merchant leaves the federation, the shared private key and the shared data can be compromised, opening a serious security problem. In addition, the coupons booklet scheme should provide measures to expel dishonest merchants from the federation, for example, using mechanisms to revoke the affiliation of merchant to the federation without compromising private data. This is a critical security issue unresolved by [7].

In the framework of the ARES project, we proposed a first approach [8] to solve the shared data problem of [7]. However, in [8], merchants could obtain coupons from honest customers and those coupons could be used by other misbehaving customers in collusion with dishonest merchants. Besides the fact that an honest customer could lose her valid coupons, she could be also accused by a dishonest party or a collusion of dishonest parties of coupon reuse. In addition, we detected problems about the fairness of the involved protocols.

Therefore, new mechanisms are needed to provide a coupons booklet scheme more secure and efficient in a *multi-merchant* scenario.

4 Providing Privacy to Coupons Booklets

In this section we describe the Privacy for Coupons Booklets scheme ($p - \mathcal{CB}$ for short), a solution to improve security and privacy from previous coupons booklets schemes, also developed under the ARES project. We start reviewing the cryptographic primitives used to provide privacy to our scheme. Afterward, we sketch all protocols which are composing the complete solution and finally, we provide proofs to demonstrate that $p - \mathcal{CB}$ is secure and improves previous security and privacy levels.

4.1 How to Provide Privacy?

Our proposal is based on the use of two cryptographic primitives: the partially blind signature and the group signature. The former is used to sign some data without

signer being able to read that data, while the latter is used to provide users with a method to operate in anonymous way on condition that they behave correctly. Below, we give a comprehensive overview about how these cryptographic primitives work.

4.1.1 Partially Blind Signature

A partially blind signature ($\mathcal{PBS}$ for short) is related to the concept of blind signatures [10] and plays a central role in cryptographic protocols where user anonymity is required, such as in electronic cash or electronic voting schemes. The main objective of blind signatures is to offer a mechanism to obtain a signature on some data which is hidden from the view of signer. But common blind signature presents an important shortcoming because the signer has no control over the blind signed parameters. To handle this limitation, $\mathcal{PBS}$ schemes [11, 12] have been proposed to allow the signer to introduce some common information in the blind signature, previously agreed between the signer and the signature requester. This feature can be assumed as a generalization of the original blind signature because it is a special case of a $\mathcal{PBS}$ where the common information is null.

In our proposal we use a simple but secure $\mathcal{PBS}$ scheme [13] based on the RSA problem. In that scheme, the *requester* (the user who requests the partially blind signature) and the *signer* (the entity who signs the request) interact through five phases:

- $Init^{\mathcal{PBS}}$. The *signer* deploys a RSA key pair, publishes his public key and selects a hash function.
- $Request^{\mathcal{PBS}}$. The *requester* asks *signer* for a partially blind signature on some hidden piece of information together with some public common information previously agreed between both parties.
- $Sign^{\mathcal{PBS}}$. Upon receipt, the *signer* signs it and sends back the result to the *signer*.
- $Extract^{\mathcal{PBS}}$. Finally, during the last stage, the *requester* proceeds to unblind the received signature obtaining this way the partially blind signature.
- $Verify^{\mathcal{PBS}}$. Anyone who has the signer's RSA public key can use this procedure to verify if a partially blind signature is valid.

4.1.2 Group Signature

The group signature schemes are an anonymous and non-repudiable multiuse credential primitive introduced by [14] to provide authenticity and anonymity to signers who belong to a group of users. This cryptographic primitive involves a group of signers, each holding a membership certificate composed by a public key for all users belonging group and an individual secret key for every signer. Any member of this group can sign messages that are publicly verifiable by anyone, hiding the identity of the actual signer within the group, i.e. the signer's identity is kept secret. However, a third party usually called the group manager, in case of any dispute or abuse, is the

only party who can trace the signature, revoking its anonymity and thus opening the real identity of signer.

In the literature there are several group signatures [15–18] but we decide to use a pairing-based group signature scheme [15] based on the Strong Diffie-Hellman (SDH) assumption, which outputs shorter signatures than other schemes with the same security level. Below, we take a look at the four procedures defined in [15]:

- $KeyGen^G(n)$. This randomized algorithm takes as input the number of members of the group (n), and outputs a group public key (pk^G), a private key of the group manager ($sk^G_{\mathcal{G}}$) and n user private keys ($sk^G_1 ... sk^G_n$).
- $Sign^G_u(pk^G, sk^G_u, m)$. Given a group public key pk^G, a user private key sk^G_u and a message m of an arbitrary length, this procedure outputs a group signature σ.
- $Verify^G(pk^G, m, \sigma)$. Given a group public key pk^G, a message m and a group signature σ, it verifies that σ is a valid group signature.
- $Open^G_{\mathcal{G}}(pk^G, sk^G_{\mathcal{G}}, m, \sigma)$. This procedure is used by the Group Manager to trace a signature to the identity of the signer. It takes a group public key pk^G, the group manager's private key $sk^G_{\mathcal{G}}$, a message m and a signature σ, and it recovers and outputs the identity of the signer.

4.2 Architecture and Protocols

Figure 2 depicts the general architecture of the $p - \mathcal{CB}$ proposal, in which we can identify four involved entities: customer ($\mathcal{C}$), merchant ($\mathcal{M}$), issuer ($\mathcal{I}$) and group manager ($\mathcal{G}$). Moreover, we define seven protocols which are responsible of all operations that the above entities can carry out among them: `Setup`, `Affiliation` and `Disaffiliation`, `Registration`, `Issue`, `Multiredeem`, `Claim` and `Refund`.

4.2.1 Setup

Both $\mathcal{G}$ and $\mathcal{I}$ must execute a setup step in order to receive requests from the other entities. $\mathcal{G}$ has to execute the $KeyGen^G$ group signature procedure to create a public group key $\left(pk^G\right)$ and the related set of secret signing keys for signers $\left(sk^G_i, \forall i \in [1, n]\right)$. Moreover, the issuer and each customer have to call to the $Init^{\mathcal{PBS}}$ procedure from the partially blind signature scheme to deploy their own RSA key pair.

4.2.2 Affiliation and Desaffiliation

Each merchant who wants to accept multicoupons issued by $\mathcal{I}$ must affiliate to this $\mathcal{I}$. This step consists on signing a simple agreement between these entities in which no sensitive information (e.g., private keys or customers information) is exchanged

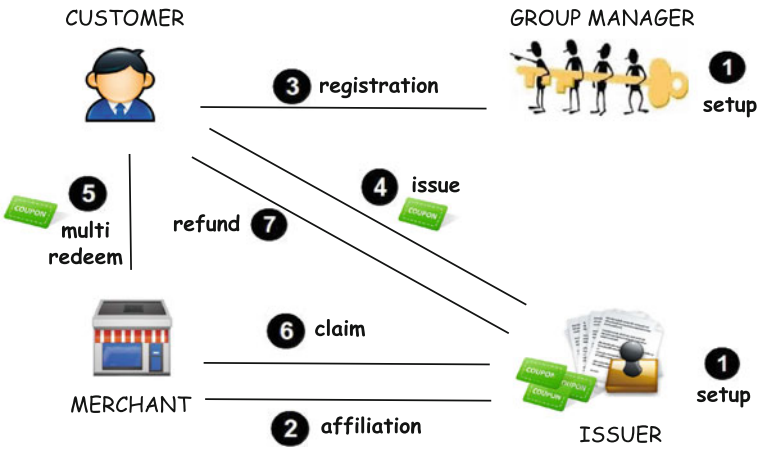


Fig. 2 $p-\mathcal{CB}$ architecture—protocols and relationships among entities

among $\mathcal{I}$ and merchants. Therefore, merchants can join and leave the affiliation under they own decision without it being cause security problems for any party.

4.2.3 Registration

Each customer who wants to use coupons from a booklet has to register with $\mathcal{G}$ using her real identity, in order to receive a group key pair $\left(pk^G, sk^G_{\mathcal{C}}\right)$ to sign on behalf of the group and to prove the fact that she actually belongs to the claimed group. During the `Registration` protocol, $\mathcal{G}$ links the real identity from $\mathcal{C}$ to the corresponding signing key in order to provide anonymity revocation in case of misbehavior.

4.2.4 Issue

Once $\mathcal{C}$ has registered to $\mathcal{G}$, she is allowed to engage with $\mathcal{I}$ to issue a signed coupons booklet, called $\mathbb{CB}$. Before explaining how the `Issue` protocol works, we review the structure of $\mathbb{CB}$ (Fig. 3). It is composed by the two following elements:

① $\mathbb{CB}_\omega$. It is the data structure that defines all coupons within a coupons booklet. Therefore, given a number of m coupons, the solution generates iteratively (by means of hash chain procedure) $2m+1$ hash identifiers from an initial random and secret *booklet seed* (ω_{seed}) up to the last hash identifier, called *booklet identifier* (ω_0). Then, each coupon (c_i) is defined as a pair of consecutive hash identifiers, where the left identifier is called *payment information* $\left(c_i^{pay} = \omega_{2i-1}\right)$ and the right one is called *proof information* $\left(c_i^{proof} = \omega_{2i}\right)$, for all $0 < i \leq m$ (i denotes the i-th coupon within the set of coupons). $\mathcal{C}$ has to keep $\mathbb{CB}_\omega$ in secret, with the exception of ω_0, which is part of the public information $\mathbb{CB}_{\mathcal{PBS}}$.

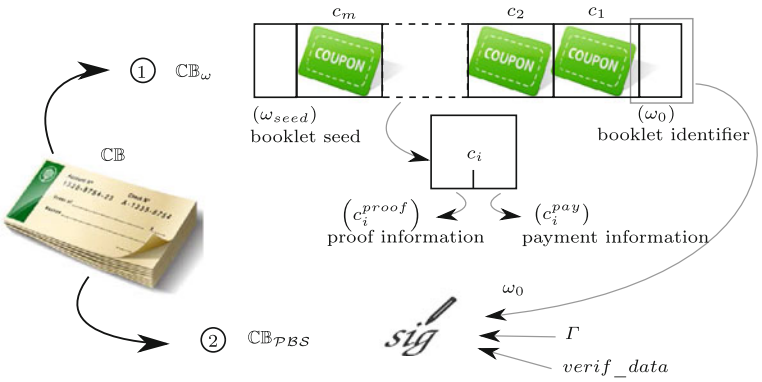


Fig. 3 $\mathbb{CB}$ structure composed by $\mathbb{CB}_\omega$ and $\mathbb{CB}_{\mathcal{PBS}}$

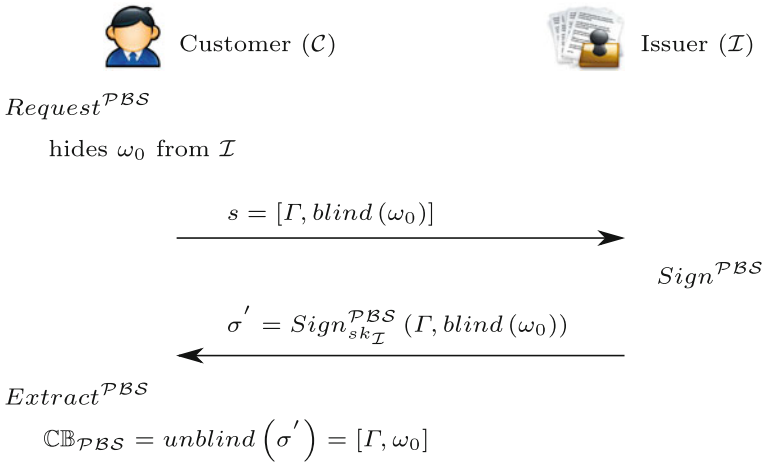


Fig. 4 `Issue` protocol flow

② $\mathbb{CB}_{\mathcal{PBS}}$. It is the partially blind signature over the booklet identifier (ω_0). The final $\mathbb{CB}_{\mathcal{PBS}}$ also conveys information, commonly agreed beforehand (Γ) between $\mathcal{C}$ and $\mathcal{I}$, which defines the coupons booklet features. These features can be different depending on the concrete application or service, but it would be common to consider parameters such as the number of coupons within the booklet, the value or discount achieved by each coupon, time marks to limit up to they are valid, etc.

Once that $\mathcal{C}$ has generated $\mathbb{CB}_\omega$, she starts the issuance process (Fig. 4) by blinding the booklet identifier (ω_0) from $\mathcal{I}$ (using $Request^{\mathcal{PBS}}$). Upon data receipt, $\mathcal{I}$ verifies the common information (Γ) and signs the received blinded data together with Γ by using his private key ($sk_\mathcal{I}$). During the last step (by means of $Extract^{\mathcal{PBS}}$), $\mathcal{C}$ completes the process without further involvement of $\mathcal{I}$. As a result, $\mathcal{C}$ unblinds the

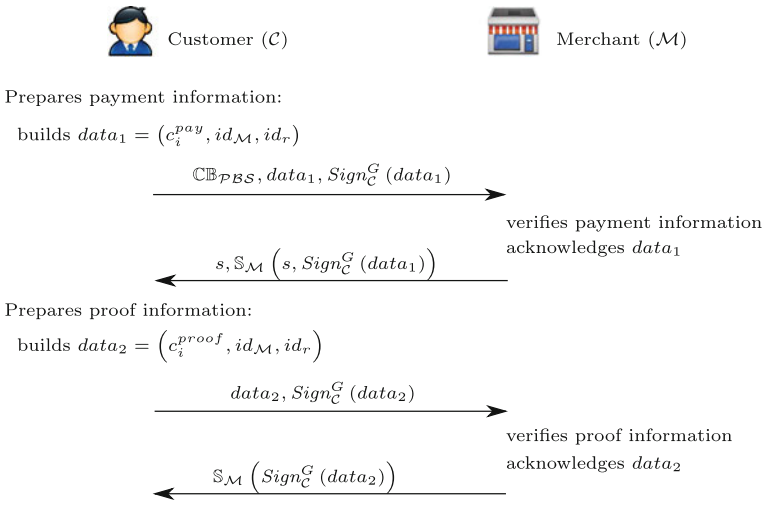


Fig. 5 `Multiredeem` protocol flow

just received signature and obtains the final $\mathbb{CB}_{\mathcal{PBS}}$. This way, $\mathcal{I}$ does not know the final face of $\mathbb{CB}_{\mathcal{PBS}}$, so $\mathcal{I}$ cannot trace the coupons booklet. The `Issue` protocol is completely anonymous because neither $\mathcal{I}$ knows any data about the identity of $\mathcal{C}$ nor the resulting $\mathbb{CB}$ contains information related to her identity. Thus, $\mathcal{C}$ can operate from this moment with her issued $\mathbb{CB}$ in an anonymous way.

4.2.5 Multiredeem

Now, $\mathcal{C}$ is ready to redeem coupons from her booklet to obtain a service from any $\mathcal{M}$ affiliated to $\mathcal{I}$ (Fig. 5). The `Multiredeem` protocol is defined by four steps, in which $\mathcal{C}$ can redeem either single or multiple coupons using a single protocol call, being this fact an important enhancement from the point of view of computing and networking efficiency.

Considering only a single coupon redemption, the `Multiredeem` protocol works as follows. $\mathcal{C}$ prepares an array of data ($data_1$) filled by the first unused *payment information* (c_i^{pay}), together with the target $\mathcal{M}$'s identifier ($id_{\mathcal{M}}$) and an identifier for the current transaction (id_r). Then, $\mathcal{C}$ group signs this data $\left(Sign_{\mathcal{C}}^{G}(data_1)\right)$ and sends it to $\mathcal{M}$, who applies some verifications, such as group signature and partially blind signature validations; he checks if the provided coupon belongs $\mathbb{CB}$; if coupon was not previously redeemed, etc. If all verifications are correct, $\mathcal{M}$ acknowledges it by means of a signature on the provided service and $data_1$ (commitment to the transaction). Then, both $\mathcal{C}$ and $\mathcal{M}$ are engaged in a similar exchange as the previous one, by sending the corresponding *proof information* (c_i^{proof}). If all verifications hold, $\mathcal{M}$ updates his stored data and acknowledges again $\mathcal{C}$ in a similar way as before. The reuse detection and avoidance can be performed checking whether any single

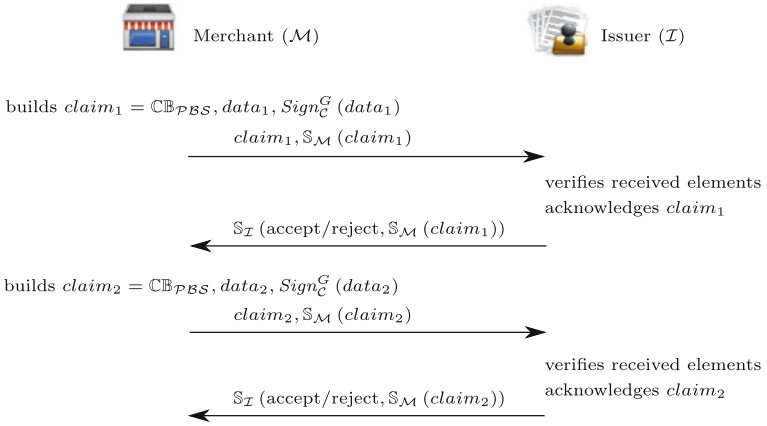


Fig. 6 `Claim` protocol flow

coupon is in the local database or in the $\mathcal{I}$'s global database (calling on-line the `Claim` protocol).

4.2.6 Claim (On-Line or Off-Line)

$\mathcal{M}$ can request $\mathcal{I}$ a money transfer to his account balance in exchange of a list of received coupons from customers (Fig. 6). $\mathcal{M}$ is allowed to claim coupons either every time a set of them is received from $\mathcal{C}$ (on-line `Claim`) or only when he has a list of received coupons from customers (off-line `Claim`).

To claim coupons, $\mathcal{M}$ has to provide $\mathcal{I}$ with information received from customers during the `Multiredeem` protocol, i.e. $data_1$ during the first step and $data_2$ during the third step of the `Claim` protocol. Upon validation, $\mathcal{I}$ acknowledges this data telling $\mathcal{M}$ whether these coupons had been used before or not (reuse avoidance). If all validations hold, $\mathcal{I}$ authorizes a deposit to the account of $\mathcal{M}$ with the right amount of money according to either the number and value of provided coupons or based on a previous agreement between both entities.

4.2.7 Refund

In case $\mathcal{C}$ is no more interested on the usage of her partially used (or completely unused) booklet, the protocol gives the opportunity to $\mathcal{C}$ to ask for a reimbursement of her remaining coupons. This protocol is optional, thus their implementation depends on the scenario as well as on the agreement among the involved entities. We omit it because it is very similar to the `Claim` protocol.

4.3 Analysis of the Provided Security

In this section we present an analysis of our $p-\mathcal{CB}$ solution to verify that it fulfills the desired security requirements. We follow the same classification of requirements as Sect. 2.2 to provide a proof analysis.

4.3.1 Basic Security Requirements

The proposed solution meets all those basic security requirements exposed in Sect. 2.2, so it offers a reliable protection against dishonest participants trying to cheat the system forging coupons or reusing them more than once. It also protects customers from their actions being linked as well as discourages the action of splitting and sharing some coupons from a booklet. We prove below all these basic security features.

Unforgeability. The number of coupons between a given booklet identifier (ω_0) and the corresponding seed (ω_{seed}) are fixed in the `Issue` protocol. The booklet identifier (ω_0) is blinded and linked to common information (data related to the number and values of coupons, expiration dates, etc.) of the $\mathbb{CB}$, and they are signed by $\mathcal{I}$. In fact, during the `Issue` protocol, the number of coupons cannot be faked. Let us suppose $\mathcal{C}$ tries to use a coupon not really included in the complete list of coupons $\mathbb{CB}_\omega$. Then, by hash chain properties, $\mathcal{M}$ can verify whether the presented coupon is included in the complete list. Note that taking a coupon, anyone can generate all coupons between that coupon and the booklet identifier (ω_0). However, the reuse of any of those coupons (see *Coupon reuse avoidance* proof below) and the over-issuing (issuing more coupons beyond ω_0 or ω_{seed}) is detected. Moreover, if an entity tries to modify the information contained in $\mathbb{CB}$, the signature validation of the $\mathbb{CB}$ allows detecting whether its contained information has been modified.

The forgery of a complete $\mathbb{CB}$ is not feasible because the issue of the $\mathbb{CB}$ requires the knowledge of $\mathcal{I}$'s private key in order to create a valid $\mathbb{CB}$. Therefore nobody, but $\mathcal{I}$, can create valid $\mathbb{CB}$ because only $\mathcal{I}$ knows her private key. Otherwise, it would mean that the security of the signature scheme has been broken, but if it is supposed secure, it only will happen with negligible probability.

Hence, $\mathcal{C}$ and $\mathcal{M}$ cannot redeem more coupons than they have been rightfully allowed, issue new coupons booklets or modify its content. So, coupons booklets are unforgeable and they cannot be modified.

Coupon reuse avoidance. $\mathcal{I}$ does not require to store any information about the $\mathbb{CB}$ during the `Issue` protocol since the $\mathbb{CB}$ was signed with her private key and it contains all the required information for its verification. However, when $\mathcal{M}$ requests the deposit to $\mathcal{I}$ in exchange to a set of received coupons, $\mathcal{I}$ stores data about these used coupons. Now, when $\mathcal{C}$ uses some coupons in a particular $\mathcal{M}$, she sends to $\mathcal{M}$ the set of data associated to those coupons to be spent, i.e., the payment and proof information along the merchant and redeem identifiers. All this information is group signed by $\mathcal{C}$ (both $\mathbb{SG}_{\mathcal{C}}\left(c_i^{pay}, id_{\mathcal{M}}, id_r\right)$ and $\mathbb{SG}_{\mathcal{C}}\left(c_i^{proof}, id_{\mathcal{M}}, id_r\right)$), and it is used

by $\mathcal{M}$ to obtain each individual coupon and check whether any single coupon has been already used, i.e., if any coupon is either in the $\mathcal{M}$'s local database or in the $\mathcal{I}$'s global database. At this point, we have to differentiate two situations:

- Assuming an honest merchant, and as explained above, a customer can try to reuse a coupon. In this case, $\mathcal{M}$ can detect it and prove the dishonest behavior of $\mathcal{C}$, because each run of the `Multiredeem` protocol is uniquely identified by the pair of identifiers $[id_{\mathcal{M}}, id_r]$. Thus, the use of the same coupons information $(c_i^{pay} - c_i^{proof})$ at the same merchant in a different `Multiredeem` run ($id_r \neq id_r^{'}$), proves customer misbehavior. Similarly, if $\mathcal{C}$ tries to reuse a same set of coupons in different merchants, the system detects it because the information (different $id_{\mathcal{M}}$ and same coupons information) is group signed by $\mathcal{C}$, and thus $\mathcal{C}$ is the dishonest participant.
- Assuming an honest customer $\mathcal{C}$, if $\mathcal{M}$ attempts to reuse a set of coupons, he must provide to $\mathcal{I}$ with the set of coupons along the pair of identifiers $[id_{\mathcal{M}}^{'}, id_r^{'}]$ signed by $\mathcal{C}$. If $id_{\mathcal{M}}^{'} = id_{\mathcal{M}}$ and $id_r^{'} = id_r$, it proves merchant misbehavior. If $\mathcal{M}$ tries to modify $id_r^{'}$ to involve $\mathcal{C}$ in an attempt of coupons reuse, the group signature on the data provided by $\mathcal{C}$ is no longer valid. $\mathcal{M}$ could collude with another $\mathcal{M}^{'}$, but in this case $\mathcal{M}^{'}$ should prove he has assigned the identifier $id_{\mathcal{M}}^{'} = id_{\mathcal{M}}$ bound to the presented coupons. Even though $\mathcal{M}$ colludes with another customer $\mathcal{C}^{'}$, $\mathcal{C}^{'}$ must sign the coupons information, and thus $\mathcal{C}^{'}$ would be charged with reuse.

Therefore, in addition to avoid the event of spending a coupon already redeemed, $\mathcal{C}$ behaving honestly can never be unfairly declared guilty of reusing a coupon or a set of them.

Unlinkability. Different coupons from different $\mathbb{CB}$ cannot be linked. Since different $\mathbb{CB}$ have different and unrelated booklet identifiers (ω_0 and $\omega_0^{'}$), two different coupons from different $\mathbb{CB}$ ($c_i \in \mathbb{CB}_{\omega}$ and $c_i^{'} \in \mathbb{CB}_{\omega}{}^{'}$, $\forall i$) cannot be linked because they are generated from an unrelated information and their corresponding booklet identifiers are different ($\omega_0 \neq \omega_0^{'}$). Therefore, we achieve a coupons booklet unlinkable scheme because different $\mathbb{CB}$ are unlinkable.

Unsplitting. According to the definition of *weak unsplittability* provided in Sect. 2.2, our scheme fits in that type of protection against splitting because customers are required to share or to use a secret. Users do not want to share sensitive information due to the fact that it could provide information to reveal their real identity (through the anonymity revocation of the group signature over a set of coupons) and thus they can be falsely accused of malicious behavior. Next, let us suppose two cases that could arise whether $\mathcal{C}$, the owner of the $\mathbb{CB}$, decides to split and share some coupons with another $\mathcal{C}'$.

- The first case could happen when $\mathcal{C}$ gives a set of group signed coupons by her to $\mathcal{C}'$, and then $\mathcal{C}'$ tries to reuse the same coupon. It is clear that in this case, $\mathcal{C}'$ does a coupon reuse which will be detected, but through the anonymity revocation mechanism of the group signature, $\mathcal{C}$ will be unfairly declared guilty of reusing a coupon.

- The second case could arise when $\mathcal{C}$ gives an unsigned set of coupons to $\mathcal{C}'$. In this case, $\mathcal{C}'$ is required to group sign the coupons with her private group key during the redeem process. Suppose that either $\mathcal{C}$ transfers a set of already used coupons or uses these coupons before $\mathcal{C}'$ uses them, then the system detects a coupon reuse event and $\mathcal{C}'$ will be accused unfairly and her identity will be revealed.

Hence, if customers want to split and share some coupons from their $\mathbb{CB}$, they assume the risk to be falsely accused of fraudulent behavior, so coupon splitting is discouraged.

4.3.2 Enhancing Customer Privacy

The $p-\mathcal{CB}$ solution not only covers basic security requirements, it even improves the customers' privacy from previous proposals with respect to merchants and issuer. In fact, it fulfills the additional privacy requirements mentioned in Sect. 2.2 which are not always present in previous solutions. We prove below how the solution meets all of them.

Anonymity of customers. $\mathbb{CB}_{\mathcal{PBS}}$ conveys two types of data: common information (expiration time, number and value of coupons), which can be read by anyone and it does not contain data about its owner; and blind data (booklet identifier ω_0). Moreover, during the issuing stage, $\mathcal{C}$ obtains the $\mathbb{CB}$ and $\mathcal{I}$ does not authenticate $\mathcal{C}$. When $\mathcal{C}$ sends either the *payment information* (c_i^{pay}) or the *proof information* (c_i^{proof}) to $\mathcal{M}$, she provides $\mathcal{M}$ with the booklet identifier signed by her group private key (sk_c^G). $\mathcal{M}$ can verify the signature using the group public key (pk^G). Therefore, neither $\mathcal{I}$ nor $\mathcal{M}$ can infer the identity of $\mathcal{C}$ since $\mathcal{G}$ is the only entity who can disclose her identity. So, customers remain anonymous in front of merchants and the issuer.

Revocation of customer's anonymity. Although forgery and reuse can be detected, the identity of $\mathcal{C}$ is not revealed without the participation of $\mathcal{G}$. As spent coupons had been signed by $\mathcal{C}$ with her group private key (sk_c^G), $\mathcal{M}$ and $\mathcal{I}$ can check the validity of each coupon. If a coupon is already used or has been forged, both $\mathcal{M}$ and $\mathcal{I}$ can ask $\mathcal{G}$ for the revocation of the anonymity of $\mathcal{C}$. $\mathcal{G}$ checks whether the reuse event was done using the proofs reported by $\mathcal{I}$. If so, $\mathcal{G}$ uses $Open^G$ to reveal the identity of $\mathcal{C}$ (see Sect. 4.1.2).

Untraceability. Although coupons within the same $\mathbb{CB}$ can be linked, these coupons cannot be traced to a particular $\mathcal{C}$. This is due to the fact that $\mathcal{C}$ does not need to reveal her identity neither in the `Issue` protocol to $\mathcal{I}$, nor in the `Multiredeem` protocol to $\mathcal{M}$. Thus, nobody can determine who spends a coupon or a set of coupons from a specified $\mathbb{CB}$. Note that even though $\mathcal{C}$ signs and sends coupons to $\mathcal{M}$ in the `Multiredeem` protocol, she uses her group private key (sk_c^G) which does not reveal any user identification. Summarizing, actions made by customers are untraceable by any party but the group manager.

Data confidentiality. Data exchanged between merchant and customer during the `Multiredeem` protocol is transferred in a confidential way, so nobody out of that secure communication channel can read the information exchanged by both parties.

Disaffiliation of merchants. We can distinguish to different types of merchants' disaffiliation depending on the reason to leave the affiliation with the issuer:

- Merchant is forced to leave the affiliation due a revocation. When a merchant misbehaves, the issuer can revoke the affiliation of the merchant easily. The issuer can include the merchant identifier $id_{\mathcal{M}}$, assigned to $\mathcal{M}$ at the affiliation stage, in an affiliation revocation list. Every time the merchant tries to claim a set of coupons, she must be authenticated by $\mathcal{I}$. Moreover, the data provided by the merchant contains that $id_{\mathcal{M}}$ signed by $\mathcal{C}$ at the redeem phase. Thus, $\mathcal{I}$ can check the revocation list and if applicable, $\mathcal{I}$ can deny the claim to $\mathcal{M}$.
- Merchant decides to leave the affiliation. $\mathcal{M}$ only needs to know $\mathcal{I}$'s public key (e, n) and the group's public key (pk^G) in order to work in the system. Therefore, merchants do not have information about customers or another sensitive information, such as the private key used to issue $\mathbb{CB}$. Moreover, if a merchant is still accepting coupons from customers after his disaffiliation, it will be under his own responsibility, because when $\mathcal{M}$ will try to claim these coupons, $\mathcal{I}$ will deny the claim due to $\mathcal{M}$ is no longer affiliated. It will be a loss for $\mathcal{M}$ not for customers. As a result, merchants can leave the system without it being compromised whether the disaffiliation is voluntary or forced.

Therefore, merchants can leave the affiliation without it causing a security flaw in any described situation.

We can conclude that customers using the $p - \mathcal{CB}$ solution remain anonymous if they behave honestly while they use their coupons. Moreover, redeemed coupons cannot be linked nor traced by merchants or the issuer to those customers who had spent them. The system also discourages customers from splitting and sharing coupons from their booklets because they can be unfairly declared guilty. In addition, the scheme protects involved honest entities from dishonest entities in such a way if a malicious customer tries to spend more coupons than those correctly issued, to reuse an already spent coupon or to redeem a fake coupon, the system detects it and the group manager reveals the cheater's identity. Similarly, a malicious merchant cannot redeem a fake coupon nor reuse a same coupon. If a merchant tries to misbehave, the system can revoke his affiliation. The system also covers the fact that a merchant would leave the affiliation by his own decision. Whichever the reason that merchants leave the affiliation, it does not cause any security flaw because merchants do not share any sensitive data.

5 Benchmarking Results

We present a real performance comparison of $p-\mathcal{CB}$ with the performance evaluation performed by Armknecht et al. in [19]. In order to compare both proposals, we have reproduced the conditions in which Armknecht et al. conducted their tests. Thus, we test $p - \mathcal{CB}$ using a Debian Linux laptop with similar features than those used in [19], i.e. a single-core CPU with 1.6GHz. The performance evaluation analyzes the time required to issue and redeem a group of five coupons ($k = 5$) as well as the

Table 2 Multi-merchant solutions—an implementation comparison

	Issuing (seconds)				Redeem (seconds)			
	$k = 5$ coupons			$k + 1$	$k = 5$ coupons			$k + 1$
	$\mathcal{C}$	$\mathcal{I}$	Total		$\mathcal{C}$	$\mathcal{M}$	Total	
[19]	–	–	4.280	0.811	–	–	33.01	6.476
$p - \mathcal{CB}$	0.023	1.182	1.205	< 0.005	0.877	1.204	2.082	< 0.02
$p - \mathcal{CB}$*	n/a	n/a	n/a	n/a	0.093	1.204	1.297	< 0.02

* applying group signature precomputation in the customer side during the multi-redeem protocol
n/a not applicable

overhead due to every new additional coupon issued or redeemed $(k + 1)$. Table 2 shows the results obtained by our proposal implementation side by side with results claimed in [19].

Talking about the `Issue` protocol, our protocol is 3.5 times faster than the same process described in [19]. In addition, the customer application only spends 23 ms of computation to issue a $\mathbb{CB}$, therefore it is in fact a lightweight protocol. Regarding the `Multiredeem` protocol, the results show that the time to redeem 5 coupons is up to 15 times faster than in [19]. Even if we apply precomputation techniques in the customer application to improve the group signature generation, the `Multiredeem` protocol can be 25 times faster than results claimed in [19]. Finally, the required time to either issue or redeem an additional coupon is negligible in our proposal, because the overhead introduced is only due to the generation and the verification of two additional hash operations. Hence, we prove that the $p - \mathcal{CB}$ solution is efficient, scalable and independent of the number of coupons in a booklet, unlike the proposal in [19].

In the test performed during the comparative, the network influence has not been considered because the tests provided in [19] are executed on a same laptop without network communication. However, we have tested our $p - \mathcal{CB}$ solution on a real deployment scenario, considering network delays. As a first approach, Fig. 7

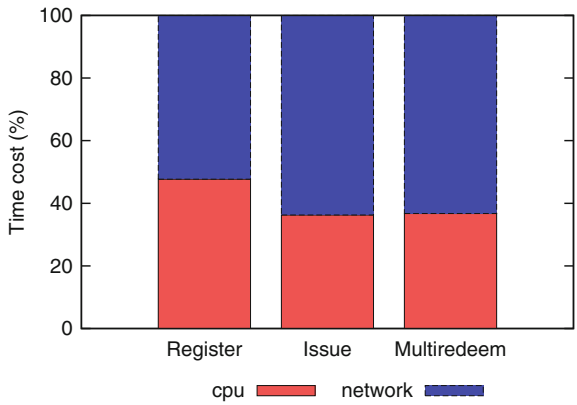


Fig. 7 Percentage of time required by computational and network operations

shows the percentage of total time required by the two main involved operations: computational and network. As we can see, network operations exceed the 50 % of the total time required for executing any of the protocols. Therefore, network influence is a very important issue that has to be considered when validating the efficiency of any m-commerce solution.

6 Conclusions

In this book chapter we reviewed the state-of-the-art of coupons booklets solutions, analyzing the previous main contributions in this field. The vast majority of proposals are related to a scenario in which issued coupons can only be redeemed by customers at the same merchant who had issued them. This is in fact a simple scenario, because merchants control the whole process from the issuance to the redemption. However, this scenario limits the usability of coupons and so this kind of solutions are not really attractive for the involved parties. As a difference from the single-merchant scenario, there are only one proposal dealing with the multi-merchant scenario which is in fact a more powerful scheme but it also present a larger complexity. Sadly, we realized that this solution does not cover all desired security and privacy requirements for this type of schemes, specially customer privacy. To handle these issues, we presented $p-\mathcal{CB}$, a coupon booklet solution ready to be applied to the multi-merchant scenario. We proved by analysis that our scheme meets all basic security requirements as well as it enhances the customer privacy and resolves some flaws from the previous proposal. It also enhances the flexibility because merchants can leave the affiliation under their own decision when they want without it being a security problem. In addition, by means of a performance evaluation and comparison, we proved that the $p-\mathcal{CB}$ scheme also improves the efficiency of the previous proposal taking into account the same computing conditions. As most relevant result, we showed that redeeming a set of coupon using $p-\mathcal{CB}$, can be up to 25 times faster than the previous proposal. In addition, the scalability is assured because the action of adding a single coupon implies a negligible overhead, as opposed to previous work.

7 Funding

This work was partially financed by the European Social Fund and the research project entitled CONSOLIDER-ARES with reference CSD2007-00004.

References

1. Chen, L., Enzmann, M., Sadeghi, A.-D., Schneider, M., Steiner, M.: A privacy-protecting coupon system. In: Proceedings of the 9th International Financial Cryptography and Data Security Conference (FC2005), Lecture Notes in Computer Science, vol. 3570, pp. 578–578. The Commonwealth Of Dominica, Springer, Roseau, Berlin, 28 Feb–3 Mar 2005

2. Nguyen, L.: Privacy-protecting coupon system revisited. In: Proceedings of the 10th Financial Cryptography and Data Security Conference (FC2006), Lecture Notes in Computer Science, vol. 4107, pp. 266–280. Paradise Cove, British West Indies, Springer, Anguilla, Berlin 27 Feb–2 Mar 2006
3. Canard, S., Gouget, A., Hufschmitt, E.: A handy multi-coupon system. In: 4th International Conference of Applied Cryptography and Network Security (ACNS2006), Lecture Notes in Computer Science, vol. 3989 , pp. 66–81. Springer, Singapore, Berlin 6–9 June 2006
4. Escalante, A.N., Löhr, H., Sadeghi, A.-R.: A non-sequential unsplittable privacy-protecting multi-coupon scheme. GI Jahrestagung **2**, 184–188 (2007)
5. Chen, L., Escalante, A.N., Löhr, H., Manulis, M., Sadeghi. A.R.: A privacy-protecting multi-coupon scheme with stronger protection against splitting. In: Proceedings of the 11th International Conference on Financial Cryptography and 1st International Conference on Usable Security (FC2007 and USEC2007), Lecture Notes in Computer Science, vol. 4886, pp. 29–44. Springer, Scarborough, Trinidad and Tobago, Berlin, 12–16 Feb 2007
6. Borrego-Jaraba, F., Garrido, P.C., Garcia, G.C., Ruiz, I.L., Gomez-Nieto, M.A.: Ubiquitous NFC solution for the development of tailored marketing strategies based on discount vouchers and loyalty cards. Sensors **13**(5), 6334–6354 (2013)
7. Armknecht, F., Escalante, A.N., Löhr, H., Manulis, M., Sadeghi, A.-R.: Secure multi-coupons for federated environments: privacy-preserving and customer-friendly. In: Proceedings of the 4th International Conference on Information Security Practice and Experience (ISPEC2008). Lecture Notes in Computer Science, vol. 4991, pp. 29–44. Springer, Sydney, Australia, Berlin, 21–23 Apr 2008
8. Isern-Deya, A.-P., Hinarejos, M.F., Ferrer-Gomila, J.-L., Payeras-Capellà, M.: A secure multicoupon solution for multi-merchant scenarios. In: Proceedings of the IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom2011), pp. 655–663. IEEE, Changsha, China, New York, 16–18 Nov 2011
9. Liu X., Xu, Q.-L.: Practical compact multi-coupon systems. In: IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS2009), vol. 3, pp. 211–216. IEEE, Shanghai, China, Washington, 20–22 Nov 2009
10. Chaum, D.: Blind signatures for untraceable payments. In: Advances in Cryptology: Proceedings of Crypto 83, vol. 82, pp. 199–203 (1983)
11. Abe, M., Fujisaki, E.: How to Date Blind Signatures. Asiacrypt' 96. Lecture Notes in Computer Science, vol. 1666, pp. 244–251. Springer, Berlin (1996)
12. Abe, M., Okamoto, T.: Provably Secure Partially Blind Signatures. CRYPTO 2000. Lecture Notes in Computer Science, vol. 1880, pp. 271–286. Springer, Berlin (2000)
13. Chien, H.-Y., Jan, J.-K., Tseng, Y.-M.: RSA-based partially blind signature with low computation. In: Proceedings of the 8th International Conference on Parallel and Distributed Systems (ICPADS2001), pp. 385–389. IEEE, Kyongju City, South Korea, Washington, 26–29 June 2001
14. Chaum, D., van Heyst, E.: Group signatures. In: Proceedings of the 10th Annual International Conference on Theory and Application of Cryptographic Techniques, EUROCRYPT'91, pp. 257–265. Springer, Brighton, UK, Berlin (1991)
15. Boneh, D., Boyen, X., Shacham, H.: Short Group Signatures. Advances in Cryptology–CRYPTO 2004. Lecture Notes in Computer Science, vol. 3152, pp. 227–242. Springer, Berlin (2004)
16. Boneh, D., Shacham, H.: Group signatures with verifier-local revocation. In: Proceedings of the 11th ACM Conference on Computer and Communications Security, CCS '04, pp. 168–177. ACM, New York, NY, USA (2004)
17. Furukawa, J.: An efficient group signature scheme from bilinear maps. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **89**(5), 1328–1338 (2006)
18. Ateniese, G., Camenisch, J., Joye, M., Tsudik, G.: A Practical and Provably Secure Coalition-Resistant Group Signature Scheme. Advances in Cryptology—CRYPTO 2000, Lecture Notes in Computer Science, vol. 1880, pp. 255–270. Springer, Berlin (2000)
19. Escalante, A.N.: Privacy-protecting multi-coupon schemes with stronger protection against splitting. Master's Thesis. Department of Computer Science, Saarland University, Germany (2008)

Smart User Authentication for an Improved Data Privacy

Vanesa Daza and Matteo Signorini

Abstract Market analysis predicts that in a few years, companies, universities, government agencies as well as common people in they daily life will increasingly adopt mobile computing systems thus increasingly enjoying the benefits of online, Internet-based services. However, such scenario will also expose user data privacy to severe attacks. This situation has led to the development of authentication approaches aimed at preventing unauthorized access to user data. Many different authentication approaches have been proposed over the last years, starting from basic password to more complex biometric solutions but all of them have proven to suffer from the same weaknesses. This issue drove us to design a solution based upon hardware intrinsic security features and aimed at guaranteeing a high level of data privacy while providing a user friendly authentication process. Our solution shows advanced features of data privacy policies definition making it a good candidate for the construction of future data privacy policies.

1 Introduction

With the advances in information and communication technology, the performance and the features of hand-held devices have rapidly increased. Formerly only a gadget to business users and aficionados or geeks, portable devices (such as smartphones or tablets) have set out to conquer the world. They are used to organize users daily lives as well as to play music, videos, games, surf the World Wide Web and take pictures using integrated cameras. They are also capable of serve health, emergency services, defense, education, banking, retailing, and other sectors benefiting from information services. Last but not least, users also store a vast array of different data on their devices, ranging from personal pictures to messages, emails, contact lists, addresses, birth-dates, music, movies and various other files.

V. Daza (✉) · M. Signorini
Department of Information and Communication Technologies, Universitat Pompeu Fabra, Barcelona, Spain
e-mail: vanesa.daza@upf.edu

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_19

Table 1 Worldwide devices shipments by segment (thousands of units)

Device type	2012	2013	2014	2017
Notebook	341,263	315,229	302,315	271,612
Ultramobile	9,822	23,592	38,687	96,350
Tablet	116,113	197,202	265,731	467,951
Mobile Phone	1,746,176	1,875,774	1,949,722	2,128,871
Total	2,213,373	2,411,796	2,556,455	2,964,783

As depicted in Table 1 Gartner market analysis [1] predicts that in 2017 there will be almost 2.7 billion portable devices and 271 million notebooks in use worldwide. Furthermore, companies, universities, and government agencies are increasingly adopting mobile computing systems and applications that allow their employees to work remotely while continuously staying connected to the organizations infrastructure.

One of the main features of such devices is the ability to use additional software applications developed by third parties and available at public catalogs. Application catalogs are widely popular as they give to the users the ability to enhance and enrich their digital experience according to their personal needs. However, as end users increasingly enjoy the benefits of such services/applications, their personal data becomes more accessible by others thus requiring a better protection.

Nowadays, people are perfectly aware about how they can take full control of their personal data, thus preventing it to fall into criminal hands.

An old Nokia Siemens Networks study [3] from 2010 and a more recent Global Research Business Network report from 2014 [2] have been conducted about the sensitivity of private data. Global Research Business Network survey on attitudes to personal and sensitive data (an excerpt of the survey is depicted in Table 2) has revealed that, on average, almost a third of US and UK citizens do not trust their domestic government. Such surveys show that, while some users continue to be indifferent about data privacy threats, more and more people are becoming conscious that safeguarding privacy is a lot more than providing adequate security. However, even though mobile device owners take steps to limit accesses to their sensitive data by apps makers, advertisers, and mobile operators, their devices can still physically fall into wrong hands. Indeed, as reported in [3], 31 % of users have already experienced the lost or stolen of their mobile device and 12 % have experienced a situation in which another person has furtively gained access to the contents of their device.

Considering the particular case of smartphones, they are no more likely to be lost or stolen than cell phones (33 % of smartphone owners against the 29 % of other cell owners) even though smartphones offer a more attractive environment for privacy invasion. Indeed, as shown in [3] both smartphone owners and basic cell phone owners reported to have experienced furtively accesses to the sensitive data stored in their device.

Table 2 Global research business network survey [2] on personal data sensitivity ranking

	Sensitive data (%)	Data (%)
National insurance number	78	17
Health records	74	19
IP or MAC address	49	38
Home address	49	40
Mobile phone location	46	41
Mobile phone number	40	49
Email address	31	54
Name	28	50
A picture of ourself	28	52
Web history	20	54
On-line purchases	19	56
Social data written about you	15	47
Social data written by you	14	46

2 Data Privacy by Encryption

The approach of data-encryption [4] can be a simple and effective way to protect personal data in all the circumstances shown so far. Encryption is said to occur when the original data (called plain-text) is transformed by a series of mathematical operations, thus generating an alternate form of the same data (called cipher-text). The goal of data encryption is to guarantee that only authorized individuals can obtain the plain-text starting from the cipher-text. During the whole encryption process the confidentiality and the integrity of plain-text data are ensured and, because of its performance, ease of implementation and cost-effectiveness, encryption is an optimal solution for securing private data at rest.

An encryption scheme requires at least three components: (i) data to be encrypted, (ii) an encryption algorithm and (iii) one or more encryption keys given as input to the encryption algorithm. Choosing the right algorithm requires the evaluation of security, performance and compliance requirements specific to user needs. While the selection of an encryption algorithm is important, protecting the keys from unauthorized access is critical. Depending on the privacy of the keys that are involved in the process, encryption algorithms can be subdivided into symmetric or asymmetric. Given the sensitivity and privacy of data stored within mobile devices, the majority of the solution aimed at protecting such data are based on symmetric algorithms. Symmetric encryption algorithms use a single key to encrypt and decrypt data and are usually the preferred approach for the privacy and security of data at rest (i.e., inactive data physically stored in any digital form). This approach is easy and fast to implement but puts the whole security in the secrecy of a single key. As such, while the selection of an encryption algorithm is important, protecting the keys from unauthorized access is critical. Cryptographic keys must be protected in storage as

improper key storage could lead to the compromise of all encrypted data.Asymmetric algorithms use two different keys. One is used for encryption and the other one for decryption. This approach is ideally suited for real-world scenarios as the secret key does not have to be shared and the risk of getting known is much smaller. However, higher speed and lower power consumption provided by symmetric algorithms are more useful.

The administration of cryptographic keys is usually performed by a key management infrastructure [5, 6] (for short, KMI). A KMI is itself composed by two elements: the storage layer and the management layer. The storage layer has the burden to securely store the plain-text keys in a digital form whilst the management layer has the burden to prevent unauthorized access to the keys. A secure way to protect cryptographic keys with a KMI is by using a hardware security module (for short, HSM). A HSM is a dedicated storage and data processing hardware element that performs all the cryptographic operations. Accesses to the HSM keys and to the HSM itself are monitored by a software-based authorization layer. Thus, with a strong encryption and authentication strategy, users can rest assured that their information assets are safe.

In the next section a survey of all the authentication solutions that have distinguished themselves in literature for originality, effectiveness and efficiency, will be introduced. Further, an analysis of all those approaches will be made and common issues will be highlighted in order to propose a new solution able to fill the gap between usability and security.

3 Authentication Approaches: State of the Art

The use of passwords is known to be ancient. Guards were used to challenge those wishing to access a restricted area to supply a key-word and would only allow a person or group of people to pass if they knew it. Nowadays, user names and passwords are widely used to regulate accesses to any kind of device or system, including mobile devices. However, traditional password schemes based on a mix of alphanumeric characters and symbols are cumbersome. Users usually disapprove their use due to the already existing amount of pass codes that they have to remember. In many cases this can even lead to bypass strategies on behalf of the user, such as choosing the same password or PIN for different applications or services or opting for passwords that are easy to remember, such as birth dates or names. The security of devices is then threatened, as users turn out to be the weakest link in the security chain. While more sensitive data are assumed to need more secure authentication methods, there is another dimension which has to be regarded: the simplicity and acceptance of respective methods. This is a critical aspect for mobile devices as users rarely use methods which are too complex or which make them feel ashamed in public.

This unappealing situation can be improved by exploiting mobile device intrinsic sensors [7, 8] and by applying unobtrusive methods for user authentication (i.e., methods that do not require either explicit attention or action by the user). The rich set

Table 3 Authentication schemes comparison

Proposed solution	Non intrusive	Progressive	Multi-factor
Clarke et al. [14]	✓	✓	–
Lin et al. [15]	✓	✓	–
Lin et al. [16]	✓	✓	–
Derawi et al. [17]	✓	✓	–
Gafurov et al. [18]	✓	✓	–
Mazhelis et al. [19]	✓	✓	✓
Conti et al. [20]	✓	–	–
Shi et al. [21]	✓	✓	✓
Riva et al. [22]	✓	✓	–
Lu et al. [13]	✓	✓	–
Zargarzadeh et al. [23]	–	–	–
Misra [24]	–	–	✓
Kang et al. [25]	✓	–	✓
Luo et al. [26]	–	–	–
Jayamaha et al. [27]	✓	–	–

of input sensors on mobile devices, including cameras, microphones, touch screens, and GPSs, enable sophisticated multimedia interactions that can be exploited for biometric authentication methods such as fingerprints [9, 10], iris [11, 12] or voice [13] recognition.

In the remainder of this section a taxonomy and further details on relevant authentication approaches will be given. In particular, all the solutions will be subdivided into macro sub-categories by considering three main aspects as: (i) intrusiveness, (ii) static nature, (iii) number of involved factors (see Table 3).

3.1 Intrusiveness

According to a recent survey [19], 60–80 % of users choose to avoid using password-based approaches because of their inconvenience thus requiring the study of non-intrusive authentication mechanisms [14]. Recently, many biometric solutions have been proposed in the literature [15–18]. Some of them, such as [17, 18] propose a gait-based authentication mechanism based on the mobile device accelerometer sensor able to recognize patterns in the movement of the user. Another solution, by Conti et al. [20], propose to use both the accelerometer and the orientation sensor to authenticate the user while answering (or placing) a phone call. This solution is based on the collection of different data sets from different sensors (such as the position in the space and other values such as pitch, roll and yaw) further analyzed by a dynamic time warping algorithm. Later on, Shi et al. [21] introduce

a non-intrusive authentication system based on four different sensors (i.e., microphone, GPS, touch screen, and accelerometer). Each sensor is then activated and used to continuously authenticate the user depending on the mobile device usage condition. As an example, the accelerometer sensor is used while the user is walking, and the touch screen sensor is used when the user is engaged in some application.

3.2 Static Nature

The problem of mobile authentication can also be studied from a completely different point of view. Rather than exploring a new authentication scheme, it is possible to study the problem of when to surface authentication and for which applications. Unlike desktops and laptops, users access mobile devices periodically or in response to a particular event. This lack of continuous interaction creates the need to authenticate with the device almost every time the users wish to use it. Even though the interaction between users and mobile devices might not be continuous, the physical contact between the user and the mobile device can be exploited for authentication purposes. As an example, if a user places his phone in his pocket after a phone call, even though the user stops interacting with it, it is still in contact with the device. When the user pulls the phone out of his pocket, authentication should not be necessary. On the other hand, if the phone lost contact with the authenticated user (e.g., left on a table), then authentication should be required. As a result, if the device is able to accurately infer the physical interaction with the authenticated user it can extend the validity of a user authentication event, reducing the frequency of such events.

This approach, named progressive authentication [22], not only significantly lowers the intrusiveness of the authentication process but also makes its effort proportional to the value of the content being accessed. If the system has strong confidence in the users authenticity, the user will be able to access any content without explicitly authenticating. If the system has low confidence in his authenticity, he will only be able to access low-value content (e.g., a weather app) and will be required to explicitly authenticate to view high-value content (e.g., email or banking). By doing so, the system provides low overhead with security guarantees that can be acceptable for a large user population, especially for those users who do not use any security lock on their mobile devices.

Progressive authentication establishes the authenticity of the user by combining multiple authentication signals. The goal is to keep the user authenticated while in possession of the device or de-authenticate the user once the user leaves it. Possible signal types could be, for example, biometric signals, behavioral signals [23] or possession signals. In combining these signals several challenges must be considered. First, most signals are produced using unreliable and discrete sensor measurements [28]. Second, certain signals may require combining readings from sources with different sampling frequencies and communication delays. As a result, most signals are not individually sufficient to determine user authenticity and when combined

they may be inconsistent as well. Finally, signals vary in strength. Some signals can provide a stronger indication of authenticity than others. As an example, some signals may be easier to fake and some other may be more discriminating. For all these reasons, signals need to be combined and cross-checked. However, manually drawing the correlations across these signals is a cumbersome job, prone to errors and inconsistencies.

Progressive authentication takes also advantage from device connectivity [24] in order to gather information from other devices owned by the user. If a user is currently authenticated into another nearby device, this information represents a strong signal of users presence. Progressive approaches can also be able to associate user identity with different confidence levels. This enables the system to depart from the all-or-nothing paradigm and allows the user to associate different protection levels to different data and applications. Users may configure their applications into appropriate security levels and specify levels by application categories (e.g., games, email, banking, etc.).

3.3 Modularity

With the rise of password cracking tools (such as [29, 30]) and faster processors, basic plain-text passwords started to be easily bypassed. Furthermore, with the advent of the cloud, many different online cracking services (such as Cloud Cracker [31]) came out making common users able to exploit the power of distributed computing for malicious activities. With this approach, as an example, 300 million password attempts can be made in as few as 20 min thus making a strong encrypted password easily corruptible.

There are basically four ways to manage passwords today and none of them are invulnerable on their own:

- Plain-Text: in this approach the password is stored in plain-text, as such, if an attacker manages to steal a plain text password file, all the private data can be steal as well;
- Basic Encryption: this approach first encrypts and then stores the password. However, advances in CPU speeds and the availability of new password cracking tools [32] still make this approach as weak as the first one;
- Random String Encryption: also known as random salt encryption [33], adds a random string to each password and then encrypts it. However, this approach still is not infallible because if the salt used is too short, or has been used more than once it is still possible to break it;
- Multiple Encryption Passes: is based on multiple encryption of the password [34].

Random string encryption and multiple encryption can both prove better data privacy and security than plain-text and basic encryption. However, the ability to utilize an incredible power in today's CPUs means that breaking the password it is just a matter of time. As such, instead of continue in the improvement of a single long

and complex password, a new approach named multi-factor authentication has been designed. Multiple factor authentication such as [25] (two-factor authentication in the particular case of only two sensors [35]), also referred to as strong authentication, consists in the joining of two or more factors such as:

- Something a user knows: this factor can be a password, a challenge question or any other secret known only by the user;
- Something a user has: this factor can consist of a small hardware device [36] (such as a smart card) used to generate a unique one-time password (such as [26]). This factor is known as a possession factor;
- Something a user is: this factor typically involves a biometric reader (such as a fingerprint [9, 10], iris [11, 12] or voice [27]) and is known as an inherence factor.

Protecting sensitive data with multi-factor approaches is one of the best policies for ensuring the safety and privacy of user data. However, such approaches still suffer from different issues thus threatening user data privacy.

4 Common Issues and Limitations

As depicted so far, regardless of the complexity of the authentication approach being used, the main weak point of the whole system is the storing of user credentials, i.e., the storing of user's private data like passwords or biometric information used in the registration step (the first authentication process). Indeed, each time a user is involved into an authentication process, data given as input must match with the original input chosen by the same user in the registration step.

Usually such original data is stored [37–39] in an encrypted way in order to avoid any malicious user to read it. However, it need to be readable at the time of the authentication process and, as such, secret keys used for its encryption/decryption need to be stored somewhere as well. Stealing such keys allows malicious users to decrypt user original input. Therefore, the rest of this section will be focused on the security of such secret keys.

Current key storage approaches can be roughly divided in two main categories: on-chip and off-chip. Off-chip key storage mechanisms are the most commonly used and are based upon memory elements, either internal or external to the device, queried by the chip as needed. Such mechanisms suffer from data eavesdropping during the transmission between chip and memories. As such, the most secure solution is to use on-chip storage like read-only memory (ROM) [40], fuse-based mechanisms [41] (e.g., poly-fuses, laser fuses, e-fuses and anti-fuses), floating-gate-based mechanisms [42] (e.g., electrically erasable programmable read-only memory (EEPROM) and erasable programmable read-only memory (EPROM) cells) and battery-backed volatile memory mechanisms [43] (e.g., battery-backed random-access memory (RAM)).

However, all such approaches have many issues as follows:

- Security: the main lack of all previous listed approaches is the permanent storage of secret keys within non-volatile memories. As such, when a user device is powered down, an adversary can use physical tools to steal the key;
- Cost: smarter solutions based on more complex elements usually require more complex construction processes that raise production costs;
- Production Time: non standard-technology device components are built on demand. As such, the request for such particular devices may cause significant production delays;
- Flexibility: ROM [40], EPROM [44] and also fuse-based memories [41] are not upgradeable in the field;
- Reliability: battery-based devices [43] have power constraints due to lifetime, temperature variations, shocks and external stresses. Furthermore, as soon as the battery is damaged or over, secret keys will be lost.

As already introduced, usually, local storage of secret keys exposes them to the risk of being stolen by malicious users able to physically access the device. Common examples of tools used to steal secret keys from devices are: electron scanning [45], laser scanning [46], confocal microscopes [47], focused ion beams [48], etc. With such tools, key bits can be glimpsed through the device thus breaking system security. It was therefore necessary to establish new affordable, but effective, security schemes not only based on key secrecy.

Given the above-mentioned issues of non-volatile storage solutions, new approaches are needed with the following features:

- keys not stored within user device;
- keys computed on-the-fly as needed;
- disposable keys;
- hardware-based keys.

Such new approaches, capable of inferring keys from device-intrinsic properties on-the-fly, overcomes many of the above limitations. Implementation of such approaches is called *Hardware Intrinsic Security* (for short, HIS) or security by *Hardware Intrinsic Properties* (for short, HIP) [49] and it can provide user authentication based on the device hardware behavior. With such new approach, user's credentials given in input are not directly stored within the device but are used as input for additional hardware computations.

More in detail, HIP advantages with respect to security, costs, time-to-market, flexibility and reliability are the following:

- Security: HIP approaches provide secret keys usage without their storage. Furthermore, when the device is powered off such keys will be no more available;
- Costs: HIP approaches do not require any additional component and, as such, production costs will remain the same;
- Production Time: HIP approaches are ready to be used with newest devices;
- Flexibility: HIP keys are field-upgradeable;

- Reliability: HIP approaches offer reliability against external influences such as temperature variations, voltage variations and humidity.

A formalization of HIP mechanisms was introduced for the first time a decade ago. First as physical one-way functions [50], then as physical random functions [51] and, finally, as physically unclonable functions [49, 52] (for short, PUFs). HIP practical relevance for security applications was immediately recognized, especially for unclonability and tamper evidence properties.

In last years, the interest in HIP has risen substantially, making them a hot topic and leading to an expansion of published results such as [53–56]. Usually, all the proposed solutions have some common properties, in fact, mostly of them perform functional operations, i.e., when queried with a given input they produce a measurable output. However, HIP functions cannot be considered as mathematical functions but as engineering functions, i.e., procedures performed upon particular (physical) systems with some given physical stimulus.

Such input stimulus sent to a hardware intrinsic properties secured device (for short, HIPD) is called *challenge* whilst the computed output is called *response*. The applied challenge given as input and the computed response produced as output are usually called a *challenge-response pair* (for short, CRP) and the relation between them is referred to as CRP behavior.

In a typical application scenario two steps are needed in order to use HIPDs. In the first phase, named *enrollment*, HIPDs are challenged with random values. Responses computed for such challenges are collected and CRPs are stored in a challenge response database (for short, CRDB). In the second phase, named *verification*, challenges from CRDBs are applied to HIPDs and computed responses are compared with the ones stored in the CRDB. If they match, then the HIPD used in the enrollment phase is the same of the verification phase and this allows user authentication by their device behavior.

4.1 Physically Unclonable Functions

The idea of using intrinsic random physical features to identify objects, systems and people is not new. In the eighties and nineties of the twentieth century, random patterns in paper and optical tokens were used for the identification of currency notes and strategic arms. In the years following the introduction of physically unclonable functions, an increasing number of new types of PUFs were proposed and the importance of PUFs unclonability and tamper evidence features for security applications was immediately recognized.

A Physically Unclonable Function (PUF) is a function that is embedded into a physical object. PUFs, by definition, are usually assumed to be *robust*, *physically unclonable*, *unpredictable* and *tamper-evident*. Informally, robustness means that the PUF has a high probability to response with the same output when challenged multiple times with the same input. Physical unclonability means that it is practical

nearly impossible to produce two PUFs that are able to produce the same output when challenged with the same input. Unpredictability means that it is unfeasible to predict the PUF behavior to an unknown challenge, even if the PUF can be trained for a certain number of times. Finally, tamper-evidence means that any attempt to physically access the PUF changes its challenge/response behavior.

There is a variety of PUF implementations. The most appealing ones, for the integration into electronic devices, are called electronic PUFs and can be built in many different ways. Delay-based PUFs are based on race conditions in integrated circuits and include arbiter PUFs [57–59] and ring oscillator PUFs [51, 52]. Memory-based PUFs exploit the instability of volatile memory cells, such as SRAM [60, 61], flip-flops [62] and latches [63, 64]. Furthermore, in [65] authors presented the concept of *Logically Reconfigurable PUFs* (for short, LR-PUFs), as a practical alternative to physically reconfigurable PUFs [49, 66, 67]. LR-PUFs amend a PUF with a stateful control logic that changes the challenge/response behavior of the LR-PUF according to its internal logical state without physically replacing or modifying the underlying PUF.

5 Hardware Intrinsic Properties for a Better Data Privacy

As already introduced, HIP approaches (such as PUFs) avoid key storage within user devices. However, in order to authenticate a device built upon HIP, accessing its CRDB is required. Such database is needed to check that responses computed by the device match the responses computed by the same device during the registration phase. This means, as shown for classical secret key storage approaches, that such CRDB must be located either within the same device to be challenged or within a remote server.

The main difference from HIP and classical secret key storage is that the secret key of the user is used to prove the identity of the user in the authentication process. However, whilst such key is computed using an algorithm in classical approaches, it is computed upon specific hardware behaviors in HIP approaches. As such, whilst an attacker in a classical approach can be able to reproduce the secret key of the victim just stealing his secret inputs (like passwords or biometric information), this is not possible with HIP approaches. In fact, challenging different devices with the same value and the same secret input will generate different outputs. However, due to HIP unpredictability, in order to check the correctness of responses, each device that wants to use HIPs needs to have access to its CRDB. As such, HIP approaches can suffer from the same issues of classical secret key storage approaches. In fact, CRDBs already contain every challenge-response pairs previously computed during the registration phase and so, an adversary able to steal such database can pretend to behave like the device of the victim.

It might then seem that the main weak point of authentication approaches based on HIP is the same of classical authentication approaches based on secret key storage but it is not. Uniqueness and unclonability properties of HIP approaches make impossible

for an adversary to steal private user information and use them to compute responses when challenged by remote services. As such, by using a HIP approach the adversary is forced to steal both victim credentials and victim's user device in order to access sensible data of the victim.

The HIP guarantees that the authentication process is not only based on user credentials, such passwords or biometric information, but is also based on a registered device owned by the user. However, even if such new authentication approach is more secure, it still requires some environment constraints. The main constraint is that CRDBs are required to be stored somewhere and contacted only at the time of user's authentication process thereby making the whole authentication scheme vulnerable to attacks.

6 APtItUDe : hArdware Based PrIvacy for User Sensitive Data

As shown in the previous section, the storage of the CRDB hinder the use of HIPDs and makes the whole authentication process vulnerable to the same attacks that we analyzed for password-based approaches. As such, we designed a new approach named APtItUDe that exploits PUF features but avoids the usage of CRDB by taking advantage of asymmetric cryptography. APtItUDe is a project aimed to guarantee a high level of data privacy while providing a user friendly authentication process. As already described in the previous sections, the wide gap that is still present between usability and security is a big challenge for nowadays researchers and developers. APtItUDe has been able to significantly reduce this gap by exploiting PUF intrinsic security properties. Indeed, it does not require any third party to grant the access to user data but it is based on hardware approaches. APtItUDe is the first authentication approach that is not tied to a specific authentication scheme thus allowing any approach to be built upon APtItUDe in order to improve data privacy. Regardless of the complexity of the *login* process, credentials given as input by the user are sent to the underneath architecture where a strong authentication key is computed. The first solution based on APtItUDe [68] introduces a novel mobile micro payment approach where all involved parties can be fully off-line.

The APtItUDe architecture is mainly based on three components, a *user credential loader*, a *key generator* and a *cryptographic element*.

- Credential Loader: this is the element that has the responsibility to read user credentials. Such credentials are then used to generate the key and can be PIN codes, passwords or even biometrical inputs. Despite all the other solutions proposed so far, such initial secret value need not necessarily be complex but it is used as a starting value to compute a strong private key;
- Key Generator: in order for the PUFs to be used in cryptographic algorithms where uniformly random and perfectly reliable keys are required, an intermediate step is required to extract a cryptographic key from PUFs. This problem is known as *secret key extraction* and it can be solved using a two-step algorithm. In the first step the

PUF is queried, thus producing an output together with someadditional information called *helper data*. In the second step, the helper data is used to extract the same output as in the first step thus making the PUF able to build cryptographic keys. It is also possible to construct a two-step algorithm guaranteeing that the key is perfectly secret, even if the helper data is publicly known. Practical instances of such kind of algorithm have been proposed in [69] and the cost of actual implementations thereof is assessed in [70]. Recently, some solutions have been proposed to correct PUF output on-the-fly thus providing the generation of secret keys within the device that is using PUFs for authentication purposes. APtItUDe uses the lightweight error correction algorithm proposed in [71]. By using such on-the-fly cryptographic key generation process, APtItUDe does not store private keys within the user device thus protecting them from malicious users and ensuring that only the right scratch card can compute its own private key with a single step each time it is needed;

- Cryptographic Element: this element is the final layer of the architecture and it is responsible to decrypt data access requests and then encrypt them back again before they leave the device. As for the key generator, also the cryptographic element is built upon PUF making it able to exploit all the features of hardware intrinsic security.

By using the cryptographic element for every data access and by encrypting user private data with the strong key computed by the key generator, APtItUDe is able to provide a strong safeguard tool for data privacy. Individual datasets can be encrypted within the memory device with a dedicated key computed on-the-fly by the key generator thus preventing malicious software to dump the memory of the device to external analysis. As such, APtItUDe is able to guarantee the following features:

- Lost/Stolen Device: if a mobile device is lost or has been stolen, user private data cannot be accessed by malicious users. In fact, PUFs embedded within the mobile device are used to compute on-the-fly the private key of the device but they still require user credentials to do it;
- Lost/Stolen User Credentials: if the PIN code or the password used by the user for the authentication process are lost or have been stolen by a malicious adversary, even if such adversary is able to steal user private data, he will not be able to read them due to the unclonability of PUFs. Because of that, the only way to have access to the user private data is to have both user credentials and the victim mobile device;
- Multiple Data Privacy Policies: by using LR-PUFs for the encryption of user private data, the same user credential can be used for different contexts/applications. In fact, by using multiple PUF configurations to identify different contexts or applications (such as business or private contexts) it is possible to use a single private user credential, e.g., a short PIN, to obtain multiple keys used by specific context/application. As an example, by using *geolocation* algorithms[1] it is possible to use such *geolabel* to identify if the user is actually at work or not.

[1] http://geohash.org/.

The Geo-label will then be used to initialize a dedicated PUF configuration further used to compute the key which will grant access to private business data. In this way, the user will just need to remember a short and easy secret credential but, depending on the Geo-location, he will be granted to access different data sets;
- Time-based Data Privacy: by exploiting LR-PUF features, it is possible to use timestamps such as different PUF configurations thus making the user able to set-up time-based access to private data. As an example, users can be able to set-up time-based data privacy policy thus making an application able to read user data just for a fixed time lapse.

7 Summary

With the advances in information and communication technology, the performance and the features of hand-held devices are rapidly increased. Market analysis predicts that in 2017 there will be almost 2.7 billion portable devices and 271 million notebooks in use worldwide. Moreover, companies, universities, and government agencies are increasingly adopting mobile computing systems and applications that allow their employees to work remotely while continuously staying connected to the organizations infrastructure. Nonetheless, as end users increasingly enjoy the benefits of online, Internet-based services their personal data becomes more exposed to attacks thus requiring a better protection. People are becoming more conscious that safeguarding privacy is about a lot more than providing adequate security. However, despite many users are taking steps to limit accesses to their sensitive data by apps makers, advertisers, and mobile operators, their devices can still physically fall into wrong hands.

Nowadays one of the most adopted solutions is the encryption of data at rest (i.e., inactive data physically stored in any digital form). The approach of data-encryption [4] can be a simple and effective way to protect personal data in all the circumstances shown so far. Symmetric encryption algorithms use a single key to encrypt and decrypt data and are usually the preferred approach for the privacy and security of data at rest. This approach is easier and faster to implement but puts the whole security in the secrecy of a single key. As such, while the selection of an encryption algorithm is important, protecting the keys from unauthorized access is critical. Cryptographic keys must be protected as improper key storage and unauthorized key accesses could lead to the compromise of all encrypted data. As such, with a strong encryption algorithm and an authentication strategy able to regulate accesses to the cryptographic keys, users can rest assured that their information assets are safe.

However, regardless of the complexity of the authentication approach being used, it came out that the main weak point of any authentication approach is the storing of user credentials (i.e., the storing of user's private data like passwords or biometric information used in the registration step) needed at user-login time. Indeed, each time a user is involved into an authentication process, data provided as input need to be matched with the original input chosen by the same user in the registration step

in order to grant the access. Usually, such registration-data is stored in an encrypted way in order to avoid any malicious user to read it. Nevertheless, such encrypted data needs to be readable at the time of the authentication process and, therefore, secret keys used for its encryption/decryption need to be stored somewhere as well.

Given the above-mentioned issues of non-volatile storage solutions new approaches are needed where keys are not stored within the user device, but rather they are hardware-based and computed on-the-fly as needed. Such new approach, capable of inferring keys from device-intrinsic properties on-the-fly, overcomes many of the above limitations. Implementation of such approaches is called *Hardware Intrinsic Security* (for short, HIS) or security by *Hardware Intrinsic Properties* (for short, HIP) and it can provide user authentication based on device hardware behavior. A formalization of HIP mechanisms was introduced for the first time a decade ago, first as physical one-way functions, then as physical random functions and, finally, as physically unclonable functions (for short, PUFs). Its practical relevance for security applications was immediately recognized, especially for unclonability and tamper evidence properties.

The novelty introduced by HIP approaches is the avoidance of key storage within the user device. However, it was not the solution to the problem but rather it shifted the problem to a lower level. Indeed, in order to authenticate a device built upon HIP, access to its *challenge-response database* (for short, CRDB) is required. Such database is needed to check that responses computed by the HIP-based device match the responses computed by the same device during the registration phase. So, the response computed at registration time can be considered the equivalent of the password-based registration-input and the response computed at login time can be considered the equivalent of the password given as input at login time. Thus, as shown for classical secret key storage approaches, this means that such CRDB must be located either within the user device or within a remote server as well. This requirement is the main constraint of HIP-based approaches and hinders the use of HIPDs making the whole authentication process vulnerable to the same attacks that we analyzed for password-based approaches.

This leads us to design a new solution capable of exploiting PUF features while avoiding the usage of stored-CRDB. The project was named APtItUDe and it was aimed at guaranteeing a high level of data privacy while providing a user friendly authentication process. APtItUDe has been able to significantly reduce the gap between usability and security by exploiting PUF intrinsic security properties underneath common authentication schemes. Indeed, it does not require any third party to grant the access to user data but it is based on hardware-based keys computed at run-time when needed. As such, APtItUDe is not tied to a specific authentication scheme but rather provides a platform capable of constructing strong data-privacy policies while allowing easy login procedures.

By using the cryptographic element for every data access and by encrypting user private data with the strong key computed by the key generator, APtItUDe is able to define individual data-sets that can be encrypted within the memory device with a dedicated hardware-based key computed on-the-fly thus preventing any malicious

semantic introspection analysis. Thanks to this, APtItUDe proved to be a strong safeguard tool with advanced features of data privacy policies definition.

References

1. Rivera, J., van der Meulen, R.: Gartner says worldwide PC, tablet and mobile phone combined shipments to reach 2.4 billion units in 2013 @ONLINE. http://www.gartner.com/newsroom/id/2408515. Accessed April 2013
2. CASRO: Global research business network study reveals widespread concern over personal data security @ONLINE. https://www.casro.org/news/162258/GRBN-Study-Reveals-Widespread-Concern-Over-Personal-Data-Security.htm. Accessed Feb 2014
3. Harjula, J.: Consumers concerned about privacy, but willing to share information with trusted telecoms operators @ONLINE. http://nsn.com/news-events/press-room/press-releases/consumers-concerned-about-privacy-but-willing-to-share-informa. Accessed Feb 2011
4. Henson, M., Taylor, S.: Memory encryption: a survey of existing techniques. ACM Comput. Surv. (CSUR) **46**(4), 53:1–53:26 (2014)
5. Martin, L.: Key-management infrastructure for protecting stored data. Computer **41**(6), 103–104 (2008)
6. Lei, S., Zishan, D., Jindi, G.: Research on key management infrastructure in cloud computing environment. In: 9th International Conference on Grid and Cooperative Computing (GCC), pp. 404–407, Nov 2010
7. Ma, Z., Qiao, Y., Lee, B., Fallon, E.: Experimental evaluation of mobile phone sensors. In: 24th IET Irish Signals and Systems Conference (ISSC), pp. 1–8, June 2013
8. Dass, S.C., Zhu, Y., Jain, A.K.: Validating a biometric authentication system: sample size requirements. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 1902–1319 (2006)
9. Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2000: fingerprint verification competition. IEEE Trans. Pattern Anal. Mach. Intell. **24**(3), 402–412 (2002)
10. Zhang, Y.-L., Yang, J., Wu, H.-T.: Sweep fingerprint sequence reconstruction for portable devices. Electron Lett **42**(4), 204–205 (2006)
11. Monro, D.M., Rakshit, S., Zhang, D.: DCT-based iris recognition. IEEE Trans. Pattern Anal. Mach. Intell. **29**(4), 586–595 (2007)
12. O'Gorman, L.: Comparing passwords, tokens, and biometrics for user authentication. Proc. IEEE **91**(12), 2021–2040 (2003)
13. Lu, H., Brush, A.J.B., Priyantha, B., Karlson, A.K., Liu, J.: SpeakerSense: energy efficient unobtrusive speaker identification on mobile phones. In: Lyons, K., Hightower, J., Huang, E.M. (eds.) Pervasive Computing, Volume 6696 of Lecture Notes in Computer Science, pp. 188–205. Springer, Berlin Heidelberg (2011)
14. Clarke, N., Karatzouni, S., Furnell, S.: Flexible and transparent user authentication for mobile devices. In: Gritzalis, D., Lopez, J. (eds.) Emerging Challenges for Security. Privacy and Trust, Volume 297 of IFIP Advances in Information and Communication Technology, pp. 1–12. Springer, Berlin Heidelberg (2009)
15. Lin, C.-C., Liang, D., Chang, C.-C., Yang, C.-H.: A new non-intrusive authentication method based on the orientation sensor for smartphone users. In: IEEE Sixth International Conference on Software Security and Reliability (SERE), pp. 245–252, June 2012
16. Lin, C.-C., Chang, C.-C., Liang, D.: A new non-intrusive authentication approach for data protection based on mouse dynamics. In: International Symposium on Biometrics and Security Technologies (ISBAST), pp. 9–14, March 2012
17. Derawi, M.O., Bours, P., Holien, K.: Improved cycle detection for accelerometer based gait authentication. In: Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), pp. 312–317, Oct 2010

18. Gafurov, D., Helkala, K., Søndrol, T.: Biometric gait authentication using accelerometer sensor. J. Comput. **1**(7), 9 (2006)
19. Mazhelis, O., Markkula, J., Veijalainen, J.: An integrated identity verification system for mobile terminals. Inf. Manage. Comput. Secur. **13**(5), 367–378 (2005)
20. Conti, M., Zachia-Zlatea, I., Crispo, B.: Mind how you answer me!: transparently authenticating the user of a smartphone when answering or placing a call. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS'11, pp. 249–259. ACM, New York, NY, USA (2011)
21. Shi, W., Yang, J., Jiang, Y., Yang, F., Xiong, Y.: SenGuard: passive user identification on smartphones using multiple sensors. In: IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp. 141–148, Oct 2011
22. Riva, O., Qin, C., Strauss, K., Lymberopoulos, D.: Progressive authentication: deciding when to authenticate on mobile phones. In: Proceedings of 21st USENIX Security Symposium, 2012
23. Zargarzadeh, M., Maghooli, K.: A behavioral biometric authentication system based on memory game. Biosci. Biotechnol. Res. Asia **10**(2), 781–787 (2013)
24. Misra, S.: A very simple user access control technique through smart device authentication using bluetooth communication. In: International Conference on Electronics, Communication and Instrumentation (ICECI), pp. 1–4. IEEE (2014)
25. Kang, J., Nyang, D., Lee, K.: Two-factor face authentication using matrix permutation transformation and a user password. Inf. Sci. **269**, 1–20 (2014)
26. Luo, S., Hu, J., Chen, Z.: An identity-based one-time password scheme with anonymous authentication. In: International Conference on Networks Security, Wireless Communications and Trusted Computing, (NSWCTC), vol. 2, pp. 864–867, April 2009
27. R.G.M.M., Jayamaha, Senadheera, M.R.R., Gamage, T.N.C., Weerasekara, K.D.P.B., Dissanayaka, G.A., Nuwan Kodagoda, G.: VoizLock—human voice authentication system using hidden markov model. In: 4th International Conference on Information and Automation for Sustainability (ICIAFS), pp. 330–335, Dec 2008
28. Moore, C., King, B.M., Vieta, W.M., Tu, X., Piemonte, P.: Calibrating sensor measurements on mobile devices, Jan 2014. US Patent 8,626,465 (2014)
29. Charoen, D.: Password security. Int. J. Secur. (IJS) **8**(1), 1 (2014)
30. Clair, L.S., Johansen, L., Enck, W., Pirretti, M., Traynor, P., McDaniel, P., Jaeger, T.: Password exhaustion: predicting the end of password usefulness. In: Proceedings of the Second International Conference on Information Systems Security, ICISS'06, pp. 37–55. Springer, Berlin, Heidelberg (2006)
31. Eli "the Computer Guy": Online hash cracking in the cloud with Cloud Cracker @ONLINE. http://www.elithecomputerguy.com/2013/03/25/online-hash-cracking-in-the-cloud-with-cloud-cracker/. Accessed Mar 2013
32. Vishwakarma, D., Veni Madhavan, C.E.: Efficient dictionary for salted password analysis. In: IEEE International Conference on Electronics, Computing and Communication Technologies (IEEE CONECCT), pp. 1–6. IEEE (2014)
33. Sharma, N., Rathi, R., Jain, V., Waseem Saifi, M.: A novel technique for secure information transmission in videos using salt cryptography. In: Nirma University International Conference on Engineering (NUiCONE), pp. 1–6, Dec 2012
34. Fujioka, A., Okamoto, Y., Saito, T.: Security of sequential multiple encryption. In: Proceedings of the First International Conference on Progress in Cryptology: Cryptology and Information Security in Latin America, LATINCRYPT'10, pp. 20–39. Springer, Berlin, Heidelberg (2010)
35. Kemshall, A.: Feature: why mobile two-factor authentication makes sense. Netw. Secur. **2011**(4), 9–12 (2011)
36. Lu, H.K., Ali, A.: Communication security between a computer and a hardware token. In: Third International Conference on Systems (ICONS), pp. 220–225, April 2008
37. Li, N., Sharif Mansouri, S., Dubrova, E.: Secure key storage using state machines. In: IEEE 43rd International Symposium on Multiple-Valued Logic (ISMVL), pp. 290–295, May 2013
38. Kalman, G., Noll, J.: SIM as secure key storage in communication networks. In: Third International Conference on Wireless and Mobile Communications (ICWMC), pp. 55–55, March 2007

39. Gallo, R., Kawakami, H., Dahab, R.: Case study: on the security of key storage on PCs. In: 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1645–1651, July 2013
40. Seok, M., Hanson, S., Seo, J.-S., Sylvester, D., Blaauw, D.: Robust ultra-low voltage ROM design. In: IEEE Custom Integrated Circuits Conference (CICC), pp. 423–426 (2008)
41. Ebrard, E., Allard, B., Candelier, P., Waltz, P.: Review of fuse and antifuse solutions for advanced standard CMOS technologies. Microelectron. J. **40**(12), 1755–1765 (2009)
42. Yoon, J.-H.: Memory properties of AI-based nanoparticle floating gate for nonvolatile memory applications. J. Korean Phys. Soc. **61**(5), 799–802 (2012)
43. Wu, M., Willy, Z.: eNVy: a non-volatile, main memory storage system. SIGPLAN Not. **29**(11), 86–97 (1994)
44. Prochnow, D.: Experiments with EPROMS. McGraw-Hill Professional, New York (1988)
45. Kratochvil, B.E., Dong, L., Nelson, B.J.: Real-time rigid-body visual tracking in a scanning electron microscope. In: 7th IEEE Conference on Nanotechnology (IEEE-NANO), vol. 28(4), pp. 498–511, April 2009
46. Korosec, M., Duhovnik, J., Vukasinovic, N.: Identification and optimization of key process parameters in noncontact laser scanning for reverse engineering. Comput. Aided Des. **42**(8), 744–748 (2010)
47. Murthy, M.S.N., Jones, M.G., Kulka, J., Davies, J.D., Halliwell, M., Jackson, P.C., Bull, D.R., Wells, P.N.T.: Infrared confocal microscope. In: IEEE Colloquium on New Microscopies in Medicine and Biology, pp. 1–2 (1994)
48. Melngailis, J.: Focused ion beam technology and applications. J. Vac. Sci. Technol. B Microelectron. Nanometer Struct. **5**(2), 469–495 (1987)
49. Sadeghi, A.-R., Naccache, D.: Towards Hardware-Intrinsic Security: Foundations and Practice, 1st edn. Springer-Verlag New York Inc, New York (2010)
50. Pappu, R., Recht, B., Taylor, J., Gershenfeld, N.: Physical one-way functions. Science **297**(5589), 2026–2030 (2002)
51. Gassend, B., Clarke, D., van Dijk, M., Devadas, S.: Silicon physical random functions. In: Proceedings of the 9th ACM conference on Computer and communications security, CCS'02, pp. 148–160. ACM, New York, NY, USA (2002)
52. Suh, G.E., Devadas, S.: Physical unclonable functions for device authentication and secret key generation. In: 44th ACM/IEEE, Design Automation Conference 2007 DAC'07, pp. 9–14, June 2007
53. van der Leest, V., Tuyls, P.: Anti-counterfeiting with hardware intrinsic security. In: Design, Automation Test in Europe Conference Exhibition (DATE), pp. 1137–1142 (2013)
54. Handschuh, H.: Hardware intrinsic security based on SRAM PUFs: tales from the industry. In: IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp. 127–127 (2011)
55. Rose, G.S., Rajendran, J., McDonald, N., Karri, R., Potkonjak, M., Wysocki, B.: Hardware security strategies exploiting nanoelectronic circuits. In: 18th Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 368–372 (2013)
56. Majzoobi, M., Koushanfar, F., Potkonjak, M.: Testing techniques for hardware security. In: IEEE International Test Conference (ITC), pp. 1–10 (2008)
57. Lee, J.W., Lim, D., Gassend, B., Suh, G.E., van Dijk, M., Devadas, S.: A technique to build a secret key in integrated circuits for identification and authentication applications. In: Symposium on VLSI Circuits, Digest of Technical Papers, pp. 176–179, June 2004
58. Ozturk, E., Hammouri, G., Sunar, B.: Towards robust low cost authentication for pervasive devices. In: Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom), pp. 170–178, March 2008
59. Lin, L., Holcomb, D., Krishnappa, D.K., Shabadi, P., Burleson, W.: Low-power sub-threshold design of secure physical unclonable functions. In: ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED), pp. 43–48, Aug 2010
60. Guajardo, J., Kumar, S.S., Schrijen, G.-J., Tuyls, P.: FPGA intrinsic PUFs and their use for IP protection. In: Proceedings of the 9th International Workshop on Cryptographic Hardware and Embedded Systems, CHES'07, pp. 63–80. Springer-Verlag, Berlin, Heidelberg (2007)

61. Holcomb, D.E., Burleson, W.P., Kevin, F.: Power-up SRAM state as an identifying fingerprint and source of true random numbers. IEEE Trans. Comput. **58**(9), 1198–1210 (2009)
62. van der Leest, V., Schrijen, G.-J., Handschuh, H., Tuyls, P.: Hardware intrinsic security from D flip-flops. In: Proceedings of the Fifth ACM Workshop on Scalable Trusted Computing, STC'10, pp. 53–62. ACM, New York, NY, USA (2010)
63. Su, Y., Holleman, J., Otis, B.P.: A 1.6pJ/bit 96 variations. In: IEEE International Solid-State Circuits Conference (ISSCC), pp. 406–611, Feb 2007
64. Kumar, S.S., Guajardo, J., Maes, R., Schrijen, G.-J., Tuyls, P.: Extended abstract: the butterfly PUF protecting IP on every FPGA. In: IEEE International Workshop on Hardware-Oriented Security and Trust (HOST), pp. 67–70, June 2008
65. Katzenbeisser, S., Koçabas, Ü., van der Leest, V., Sadeghi, A.-R., Schrijen, G.-J., Schröder, H., Wachsmann, C.: Recyclable PUFs: logically reconfigurable PUFs. In: Proceedings of the 13th International Conference on Cryptographic Hardware and Embedded Systems, CHES'11, pp. 374–389. Springer-Verlag, Berlin, Heidelberg (2011)
66. Kursawe, K., Sadeghi, A.-R., Schellekens, D., Skoric, B., Tuyls, P.: Reconfigurable physical unclonable functions—enabling technology for tamper-resistant storage. In: Proceedings of the 2009 IEEE International Workshop on Hardware-Oriented Security and Trust, HST'09, pp. 22–29. IEEE Computer Society, Washington, DC, USA (2009)
67. Lim, D., Lee, J.W., Gassend, B., Suh, G.E., van Dijk, M., Devadas, S.: Extracting secret keys from integrated circuits. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **13**(10), 1200–1205 (2005)
68. Daza, V., Di Pietro, R., Lombardi, F., Signorini, M.: Fully off-line secure credits for mobile micro payments. Internal report
69. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. SIAM J. Comput. **38**(1), 97–139 (2008)
70. Maes, R., Tuyls, P., Verbauwhede, I.: Low-overhead implementation of a soft decision helper data algorithm for SRAM PUFs. In: Proceedings of the 11th International Workshop on Cryptographic Hardware and Embedded Systems, CHES'09, pp. 332–347. Springer-Verlag, Berlin, Heidelberg (2009)
71. Yu, M.-D.M, M'Raihi, D., Sowell, R., Devadas, S.: Lightweight and secure PUF key storage using limits of machine learning. In: Proceedings of the 13th International Conference on Cryptographic Hardware and Embedded Systems, CHES'11, pp. 358–373. Springer-Verlag, Berlin, Heidelberg (2011)

Part VII
User Privacy: Web Search Engines

Multi-party Methods for Privacy-Preserving Web Search: Survey and Contributions

Cristina Romero-Tris, Alexandre Viejo and Jordi Castellà-Roca

Abstract Web search engines (WSEs) locate keywords on websites and retrieve contents from the World Wide Web. To be successful among its users, the WSE must return the results that best match their interests. For this purpose, WSEs collect and analyze users' search history and build profiles. Although this brings immediate benefits to the user, it is also a threat for her privacy in the long term. Profiles are built from past queries and other related data that may contain private and personal information. Consequently, researchers on this field have developed different approaches whose objective is to avoid this privacy threat and protect users of WSEs. One way to classify the existing alternatives is between single-party and multi-party. The former approach allows users to protect their privacy in front of the WSE without requiring the cooperation of others. The latter requires that a group of users or entities collaborate in order to protect the privacy of each member of the group. This work focuses on multi-party schemes. First, current solutions in this field are surveyed, their differences are analyzed and their advantages (and shortcomings) are stressed. Finally, our own contributions to this area are presented and evaluated.

1 Introduction

A web search engine (WSE) is a tool designed to find and retrieve information from the Internet. There are many examples of WSEs in the market, such as Google, Bing, AOL, etc.

When a user wants to search a term in a WSE, she types the keywords in a search bar and submits her query. Then, the WSE applies information retrieval techniques to select and rank the results. The better these results are ranked according to each

C. Romero-Tris (✉) · A. Viejo · J. Castellà-Roca
Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Spain
e-mail: cristina.romero@urv.cat

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_20

user's preferences, the more successful the WSE will be. For this reason, WSEs collect and analyze users' search history in order to build profiles. Consequently, a profiled user who submits a certain query will receive the results which are more interesting for her, in the first positions.

Although profiling leads to a useful service, it can also raise privacy concerns. The logs that WSEs store and analyze to build profiles may contain sensitive data that can be combined to disclose information of a certain individual. Consequently, it is important to control how this information is managed. However, incidents in the past have shown that WSEs cannot be trusted in this matter. In 2006, AOL released a file with twenty million queries generated by its users [1]. This incident had serious consequences since personally identifiable information was present in many of the queries. Another example of these incidents is the subpoena that Google suffered in 2006 [2], where the Justice Department of U.S.A. tried to compel Google to provide millions of Internet search records. Other privacy risks of query logging (e.g., disclosure for marketing purposes) are described in [3].

As a response to these privacy risks, researchers in this area have developed some alternatives that protect users' privacy in web search. Some works (e.g., [4, 5]) classify these alternatives into two groups: single-party protocols and multi-party protocols. Single-party protocols (such as [4, 6–11]) submit machine-generated queries combined with the queries of the user. The challenge of single-party schemes is to generate queries that cannot be distinguished from human-generated queries. In a multi-party protocol, a group of users is created. Then, a user asks another component of the group to submit her query and to send back the results.

The matter of privacy-preserving in web search is too vast to be covered in a single work. There are many privacy-preserving alternatives, and the techniques applied in single-party and multi-party are very different. Moreover, single-party approaches are already at an advanced state of development, while multi-party approaches still have room for improvement. For all these reasons, this work specifically focuses on multi-party protocols.

2 Previous Work on Multi-party Schemes

In this section, we survey the previous research done in multi-party private web search. We summarize the major contributions to the field, and review the proposals on which our work is based.

In a multi-party protocol, a group of users is created. Then, a user asks another component of the group to submit her query and to send back the result. We can classify multi-party protocols according to the way users are grouped. There are multi-party protocols that use static groups, where the same members participate in every execution of the protocol. On the other hand, in multi-party protocols with dynamic groups, a user is grouped with different members, every time that she executes the protocol.

The main advantage of using static groups instead of dynamic is that groups are only generated once. This means that multi-party protocols with static groups are usually faster, since they do not need an initial phase to generate the group every time that the user wants to submit a query. On the other hand, multi-party protocols with dynamic groups need to run a setup phase in every execution of the protocol. This phase often requires a central entity that provides the users with the information to contact the other members of the group. Sometimes, this central node can represent a bottleneck.

The main weakness of multi-party protocols with static groups is that they are more vulnerable in front of a targeted internal attack. These attacks occur when a malicious user tries to learn the queries of a specific user. If the malicious user succeeds, a profile of the victim can be built. With dynamic groups, there is a very small probability that an attacker is grouped with her victim twice. Furthermore, in order to build a complete profile of her victim, the attacker needs to join her in the same group several times. This requirement reduces even more the success probability of this kind of attacks when using dynamic groups. Note that these attacks are easier using static groups because, if the attacker is grouped with her victim, this link will be maintained for all the executions.

Next, we describe the main contributions in the literature to multi-party protocols with static and dynamic groups.

2.1 Multi-party Protocols with Static Groups

The works presented in [12, 13] propose two multi-party protocols where the groups are static. In those schemes, the same members participate in every execution of the protocol. Accordingly, the WSE could build a profile from a group but not from a specific user.

In [12], the authors present a system named Crowds. This system puts users into a large group (the crowd) where they submit requests on behalf of other members. Users in the system are represented by processes called "jondos". Jondos are assigned to a "crowd" with other jondos by an administrative process called "blender". The blender is also responsible for informing new jondos of other members of the crowd and for informing all the jondos when a user joins/leaves the crowd. Besides, every node has a direct link with each other node of the network (the topology is a complete graph). Communication through the link is encrypted using a key only known by the two jondos (point-to-point cryptography).

When a user wants to make a request, she establishes a random path through the network. For this purpose, she randomly picks another jondo and forwards the request to it. That jondo then flips a coin with a forwarding probability $p_f > 0.5$. Depending on the outcome of the coin flip, the jondo either randomly selects another jondo to which the request will be forwarded, or it forwards the request to the intended web server (the WSE). In this way, some node eventually submits the query to the WSE.

Then, the results are forwarded towards the original node, following the reversed path that the query used.

The security of the system relies on the fact that when a jondo receives a request, it does not know whether the sender was the original requester of or whether it was forwarding the request for another jondo. Note that there is a tradeoff between the forwarding probability p_f and the length of the path and, hence, the query delay. The more jondos the query visits, the higher the query delay is and the lower the performance for the user is.

Nevertheless, Crowds suffers from the following problems:

- The blender represents a bottleneck in the overall system performance.
- The use of point-to-point encryption and a complete graph require that each node stores as many symmetric keys as nodes are in the network. In addition to that, each hop requires one encryption and one decryption. This means that every hop introduces overhead in the system.
- Like in the anonymity networks, Crowds only protects the transport of the data. Users are responsible for hiding their private information.
- Building a profile from a group is only possible if the members of the crowd share the same interests.
- This scheme is weak against the *predecessor attack* [14]: To attack Crowds, a number of attackers may simply join the crowd and wait for paths to be reformed. Each attacker can log its predecessor after each path reformation. Let us consider a user U who wants to submit several queries. Due to the randomly distribution of the queries among all the nodes of the network, U will forward almost all her queries to the rest of the users. As a result, several reformations will happen and U will appear in all of them. Therefore, the attackers will log U much more often than any other node. After a large number of path reformations, it will become clear that U is the user who is generating the queries.

The four first problems are solved in the proposal presented in [13]. This work is based on the same principles applied in [12]. The difference with Crowds is that it can be applied to already developed social networks (e.g., Facebook). This fact gives [13] a big advantage over [12] in terms of deployability and query delay. Nevertheless, the predecessor attack is still a weakness in both systems.

In the scenario of [13], the user belongs to a group which is formed by her friends in the social network. Consequently, it is more likely that the users inside the group share interests. Similarly to [12], when a user wants to submit a query to the WSE, she forwards it to a pseudorandomly selected node among her friends. The node that receives the query can either submit or reforward it to another node. At the end, some node submits the query to the WSE and sends back the response, following the path used by the query, conversely.

Another advantage of this scheme is that the central node and the point-to-point shared keys are no longer needed. In addition, each hop does not have to decrypt and encrypt the query, which makes the process faster. In the simulations performed by the authors, their scheme achieves the lowest delay (3.9 s) in the current literature.

The scheme presented in [15] presents a new version of the protocol presented in [13], with some improvements. Firstly, it increases the level of privacy obtained by the users, by equitably distributing the queries in a path of length two. This means that it considers not only the queries submitted by direct neighbours but also the neighbours of the neighbours. Consequently, the source of a query is better hidden than in [13], since it is hidden among a group with a path length of two.

However, both schemes ([13, 15]) are affected by the problem of using static groups. The group of a particular user is formed by her contacts in the social network. Consider the case where one of the contacts of the user is a dishonest party trying to keep track of her queries. As long as the link between them exists, the attacker will receive, with a certain probability, queries that belong to her victim. Furthermore, in some social networks, we can assume that the attacker and the victim share a relationship that gives to the attacker a certain knowledge of her victim. Therefore, in such cases, it would be easier for the attacker to guess when the victim is forwarding a query on behalf of another user, and when she is sending her own query.

Another example of a multi-party system is the system proposed in [16]. This scheme uses memory sectors which are shared by a group of users. These users employ 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, a simple wiki-like collaborative environment can be used to implement a shared memory sector. This scheme has the following drawbacks when applied in a WSE scenario:

- 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.
- The best response time achieved by this proposal is 5.84 s. However, the authors do not include the network delay in this time. According to that, the final response time is expected to be clearly above 5.84 s but the exact value is not specified.

However, this scheme is also affected by the problem of using static groups. The group of a particular user is formed by her contacts in the social network. Consider the case in which one of the contacts of the user is a dishonest party trying to keep track of her queries. As long as the link between them exists, the attacker will receive, with a certain probability, queries that belong to her victim. Furthermore, in some social networks, we can assume that the attacker and the victim share a relationship that gives to the attacker a certain knowledge about her victim. Therefore, in such cases, it would be easier for the attacker to guess when the victim is forwarding a query on behalf of another user, and when she is sending her own query.

2.2 Multi-party Protocols with Dynamic Groups

In [17], the authors propose a multi-party protocol named Useless User Profile (UUP). The basic idea beneath this system is that a central node puts users into dynamic groups in which they securely exchange their queries. As a result, each user submits a query from one of her partners and not her own and, hence, she obtains a distorted profile.

More specifically, there is a central node that groups n users that intend to submit a query. Those n users execute a protocol where each user U gets a query from one of the other $n-1$ users. The protocol requires the user U not to know the source of the received query. For this purpose, all the queries are first shuffled and then distributed. This shuffling is performed using encryptions, re-maskings and permutations.

After the distribution of the queries, each user submits the received query to the WSE. The response is then broadcast to all the members of the group. Each user selects only her answer and discards the rest.

The UUP protocol achieves a query delay of 5.2 s. This time significantly outperforms similar proposals. However, the authors leave two points of improvement as future work:

1. *Reduce the query delay, which is still high.*
   A 5.2 s query delay is acceptable for occasional use. However, for a frequent use, it is necessary to minimize the query delay. This would increase the level of satisfaction of the users with the application.
2. *Prevent a dishonest user from obtaining the same level of privacy even if she does not follow the protocol.*
   In the UUP protocol, a user could behave selfishly and obtain the same results as an honest user. Even if the selfish user does not decrypt a query, submit it to the WSE and broadcast it to the rest of the users, she can still receive the response for her query. As a result of this behavior, there would be one honest user who would not get the results that she was expecting. This is a vulnerability of the protocol that must be addressed.

Besides, the UUP protocol has a major disadvantage. It is not secure in presence of malicious internal users. The protocol assumes that the users follow the protocol and that there are no collusions between two entities. The authors argue that this assumption is reasonable since the objective of the protocol is to protect users from WSE profiling.

In [18], Lindell and Waisbard consider a scenario which is similar to the scenario proposed in [17]. However, they argue that the level of security that the UUP protocol provides is not sufficient. They identify some attacks that malicious internal users can perform in order to learn the queries of another member of the group. Hence, they modify the UUP protocol in order to be resilient against these attacks. We next describe the main differences between the UUP protocol and the modification that Lindell and Waisbard propose.

The main difference is that their solution is based on a concept called *private shuffle*. As previously mentioned, in the UUP protocol, a shuffle is performed

applying encryptions, re-maskings and permutations; and the inputs of the shuffle are the queries of the user. However, in a *private shuffle*, the inputs are already encrypted versions of the queries. Informally, this means that, during the protocol, queries are protected under a double encryption.

Both proposals perform the shuffle in a similar way. After the shuffle, the outputs are decrypted. In the UUP protocol, this means that the users obtain the cleartexts of the queries. However, in the modification presented by Lindell and Waisbard, this only means to remove the outermost layer of encryption and, hence, queries are not visible yet. Each user then checks that her encrypted query is one of the outputs of the shuffle. In this case, she sends *true* to the other members. Otherwise, she sends *false*. If all parties send *true*, they can then proceed to decrypt the inner layer of encryption of the queries. This way, the proposed modification ensures that no malicious behavior has occurred during the shuffling.

Nevertheless, the drawback of this proposal is that it uses expensive cryptographic tools (i.e., double encryptions) that introduce an unaffordable query delay. Although their work does not include any simulation, the authors remark that executing their protocol is, at least, twice as expensive as the UUP protocol.

3 Contributions and Organization of This Document

All the proposals presented in Sect. 2 have some advantages and some disadvantages. In general, they show that achieving the right balance between protecting privacy and offering an affordable query delay is a difficult challenge.

The contributions that we present in this document focus on multi-party protocols with dynamic groups. We chose dynamic groups over static ones because we consider that static groups are more vulnerable in front of a targeted internal attack. Besides, the UUP protocol is a multi-party protocol with dynamic groups that leaves the two points of improvement described in Sect. 2 as future work. Therefore, based on an analysis of this protocol, we present our two contributions in this field:

- *Contribution-1*: A protocol that improves the *query delay* of the UUP protocol. We consider a scenario where users need to submit many queries quite frequently. The feature that we intend to maximize is the speed of the system. Regarding privacy in front of the WSE, similarly to the UUP protocol, we require the WSE not to be able to distinguish the real source of a submitted query. Hence, our scheme enables users to privately submit queries and receive the results in a reasonable amount of time.
- *Contribution-2*: A protocol that improves the *level of security* of the UUP protocol. Here, we consider a more hostile scenario with stronger attackers than in the UUP protocol. In this case, obtaining the lowest query delay is not an essential requirement. Instead, we study a scenario where all the entities that participate in the protocol are potential attackers. We analyze the possible attacks that honest users may suffer, and propose a solution to prevent them.

This document is organized as follows: Sect. 4 introduces the background and notation required to understand the new proposals. Section 5 presents the first contribution. Section 6 presents the second contribution. Finally, Sect. 7 includes the conclusions of the work.

4 Background and Notation

This section introduces the cryptographic background, assumptions and definitions required to understand the contributions described in subsequent sections.

First of all, the n-out-of-n threshold ElGamal encryption is explained (Sect. 4.1). Then, a permutation network named OAS-Benes is introduced in Sect. 4.3. Finally, two zero-knowledge proofs called Plaintext Equivalence Proof (PEP) and Disjunctive Plaintext Equivalence Proof (DISPEP) are described in Sects. 4.4 and 4.5, respectively.

4.1 n-out-of-n Threshold ElGamal Encryption

In cryptographic multi-party protocols, some operations must be computed jointly by different users. In an n-out-of-n threshold ElGamal encryption [19], n users share a public key y and the corresponding unknown secret key α is divided into n shares α_i. By using this protocol, a certain message m can be encrypted using the public key y and the decryption can be performed only if all n users collaborate in the decryption process. Key generation, encryption and decryption process are next described.

Key generation First, a large random prime p is generated, where $p = 2q + 1$ and q is a prime number, too. Also, a generator g of the multiplicative group $\mathbb{Z}_q^*$ is chosen.

Then, each user generates a random private key $\alpha_i \in \mathbb{Z}_q^*$ and publishes $y_i = g^{\alpha_i}$. The common public key is computed as $y = \prod_{i=1}^{n} y_i = g^{\alpha}$, where $\alpha = \alpha_1 + \cdots + \alpha_n$.

Message encryption Message encryption can be performed using the standard ElGamal encryption function [20]. Given a message m and a public key y, a random value r is generated and the ciphertext is computed as $E_y(m, r) = c = (c1, c2) = (g^r, m \cdot y^r)$

Message decryption Given a message encrypted with a public key y, $E_y(m, r) = (c1, c2)$, user U_i can decrypt that value as follows:

Each user $j \neq i$ publishes $c1^{\alpha_j}$. U_i can then recover message m in the following way:

$$m = \frac{c2}{c1^{\alpha_i}(\prod_{j \neq i} c1^{\alpha_j})}$$

This decryption can be verified by each participant by performing a proof of equality of discrete logarithms [21].

- Message partial decryption. An alternative for the message decryption described above is the partial decryption, which allows a group of users to jointly decrypt a ciphertext.
  Similarly to the normal message decryption, given a message encrypted with a public key y, $E_y(m, r) = (c1, c2)$, U_i employs her private key α_i to partially decrypt the ciphertext as follows:

$$c2' = \frac{c2}{c1^{\alpha_i}}$$

  The result of this operation is another ciphertext denoted as $E_{y'}(m, r) = (c1, c2')$. In this case, the ciphertext is encrypted with a public key $y' = g^{\alpha_{(i+1)}+\cdots+\alpha_n}$ which no longer contains the private key belonging to U_i.

4.2 ElGamal Re-masking

The re-masking operation performs some computations over an encrypted value. In this way, its cleartext does not change but the re-masked message is not linkable to the same message before re-masking.

Given an ElGamal ciphertext $E_y(m, r)$, it can be re-masked by computing [22] $E_y(m, r) \cdot E_y(1, r')$, for $r' \in \mathbb{Z}_q^*$ randomly chosen, and where $\cdot$ stands for the component-wise scalar product (ElGamal ciphertext can be viewed as a vector with two components). The resulting ciphertext corresponds to the same cleartext m.

4.3 Optimized Arbitrary Size (OAS) Benes

A Benes permutation network (PN) is a directed graph with N inputs and N outputs, denoted as $PN^{(N)}$. It is able to realize every possible permutation of N elements.

A Benes PN is composed of a set of 2×2 switches. These switches have a binary control signal $b \in \{0, 1\}$ which determines the internal state and, hence, the output. The two possible states of a 2×2 switch are depicted in Fig. 1a.

The problem with a Benes PN is that the size of the network must be a power of 2. in order to have an Arbitrary Sized (AS) Benes network, it is necessary to introduce a 3×3 network, like Fig. 1b shows. By using 2×2 switches and 3×3 networks recursively, it is possible to construct a network of any size.

Optimized Arbitrary Size (OAS) Benes is an extension of AS Benes that reduces the number of necessary switches in the network. The way of constructing the OAS-Benes depends on the parameter N:

- If N is even, the OAS-Benes $PN^{(N)}$ is built recursively from two even OAS-Benes of $\frac{N}{2}$-dimension called sub-networks. The sub-networks are not directly connected

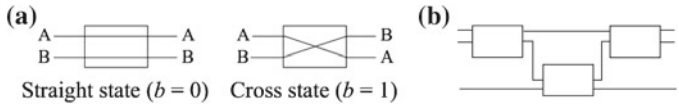


Fig. 1 Basic elements of an OAS-Benes

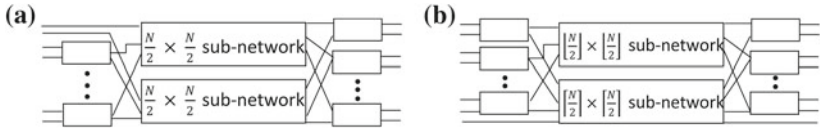


Fig. 2 Construction of OAS-Benes

to the inputs and outputs. Instead, they are connected to $N-1$ input-output switches, as Fig. 2a shows.

- If N is odd, the OAS-Benes $PN^{(N)}$ is composed by an upper $\lfloor \frac{N}{2} \rfloor$ even OAS-Benes, and a lower $\lceil \frac{N}{2} \rceil$ odd OAS-Benes. The sub-networks are not directly connected to the inputs and outputs. In this case, the first $N-1$ inputs are connected to $\lfloor \frac{N}{2} \rfloor$ switches, and the first $N-1$ outputs are connected to $\lfloor \frac{N}{2} \rfloor$ switches. Figure 2b illustrates this construction.

According to the way that an OAS-Benes is constructed, it is possible to account the minimum number of switches required to satisfy a permutation of N elements. The formula to calculate the minimum number of switches is:

$$S(N) = \begin{cases} (N-1) + 2 * S(\frac{N}{2}) & \text{if } N \text{ is even} \\ 2 * \lfloor \frac{N}{2} \rfloor + S(\lceil \frac{N}{2} \rceil) + S(\lfloor \frac{N}{2} \rfloor) & \text{if } N \text{ is odd} \end{cases}$$
$$\text{where } S(1) = 0,\ S(2) = 1,\ S(3) = 3$$

Multi-party OAS-Benes OAS-Benes can be used to perform a joint permutation. This means that the switches of the OAS-Benes can be distributed among a group of n users trying to realize a permutation of N inputs. However, this must be done in a way that no user knows the overall permutation between the inputs and the outputs.

According to [23], a secure permutation (where no user knows the overall permutation) requires minimally t OAS-Benes $PN^{(N)}$, where t depends on the minimum number of honest users that the system requires. The t OAS-Benes $PN^{(N)}$ are fairly divided into n adjacent stages. Then, stage i (for $i \in 1, \ldots, n$) is assigned to user i.

In order to obtain a secure permutation, the condition that must be satisfied is that honest users control, at least, $S(N)$ switches. For example, consider a scenario with $n = 6$ users, $N = 8$ inputs and, at least, $\lambda = 3$ honest users. The number of switches of one OAS-Benes $PN^{(8)}$ is $S(8) = 17$. According to [23], the $\lambda = 3$ honest users must control 17 or more switches. This means that every user must control $\lceil \frac{17}{3} \rceil = 6$ switches. Therefore, the scheme needs at least (6 *switches per user times* 6 *users*) =

36 switches that will be fairly divided among the n users. Consequently, the system requires $t = \lceil \frac{36}{17} \rceil = 3$ OAS-Benes $PN^{(8)}$.

We propose formula 1 in order to calculate the number of OAS-Benes required in a scheme with n users, N inputs, and λ honest users.

$$t = \left\lceil \frac{n \cdot \left\lceil \frac{S(N)}{\lambda} \right\rceil}{S(N)} \right\rceil \tag{1}$$

4.4 Plaintext Equivalence Proof

Plaintext Equivalence Proof (PEP) [24] is an honest-verifier zero-knowledge proof protocol based on a variant of the Schnorr signature algorithm [25]. The purpose of this protocol is to prove that two different ciphertexts are the encryption of the same message.

Two ElGamal ciphertexts $(c1_a, c2_a) = (g^{r_a}, m_a \cdot y^{r_a})$ and $(c1_b, c2_b) = (g^{r_b}, m_b \cdot y^{r_b})$ for some $r_a, r_b \in \mathbb{Z}_q^*$ are plaintext equivalent if $m_a = m_b$. Let:

- $\alpha = r_a - r_b$
- $k = H(y, g, c1_a, c2_a, c1_b, c2_b)$, where $H(\cdot)$ is a hash function.
- $G = g \cdot y^k$
- $Y = \frac{c1_a}{c1_b} \cdot (\frac{c2_a}{c2_b})^k = (g \cdot y^k)^\alpha$

In order to prove that $(c1_a, c2_a) \equiv (c1_b, c2_b)$, the prover must demonstrate knowledge of α by executing the protocol of Fig. 3.

4.5 Disjunctive PEP

Disjunctive PEP (DISPEP) [24] is an extension of the PEP protocol. In this case, a prover demonstrates that one of two different ciphertexts is a re-masked version of another ciphertext.

Let $(c1_a, c2_a) = (g^{r_a}, m_a \cdot y^{r_a})$ and $(c1_b, c2_b) = (g^{r_b}, m_b \cdot y^{r_b})$ be two different ElGamal ciphertexts. Then, one of them is a re-masking of another ciphertext $(c1, c2) = (g^r, m \cdot y^r)$ for some $r_a, r_b, r \in \mathbb{Z}_q^*$ if $m_a = m$ or $m_b = m$. For $i \in \{a, b\}$,let:

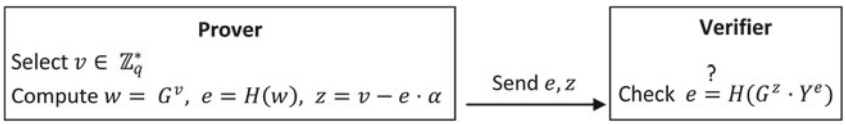


Fig. 3 PEP protocol

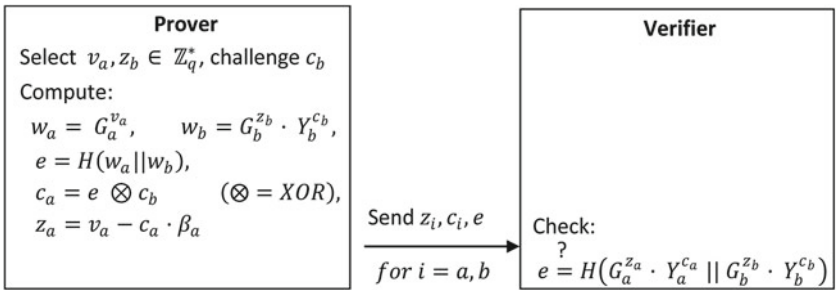


Fig. 4 DISPEP protocol

- $\beta_i = r - r_i$
- $k_i = H(y, g, c1, c2, c1_i, c2_i)$
- $G_i = g \cdot y^{k_i}$
- $Y_i = \frac{c1}{c1_i} \cdot (\frac{c2}{c2_i})^{k_i} = (g \cdot y^{k_i})^{\beta_i}$

In order to prove whether $m_a = m$ or $m_b = m$, the prover must demonstrate knowledge of β_i by executing the protocol of Fig. 4. Without loss of generality, in Fig. 4 we assume that the prover is showing $m_a = m$.

5 Contribution-1: Improving Query Delay

This section describes a modification of the work presented in [17], the UUP protocol. On one hand, our system optimizes some steps of the protocol to reduce the delay of each query. On the other hand, these changes incentivize every user to follow the protocol in order to protect their own privacy. This proposal was published in [26].

5.1 Protocol Description

Group setup The user U_i who wants to submit a query to the WSE, contacts the central node, requesting to be included in a group. The central node is listening to user requests. Once it has n requests, a group $\{U_1, \ldots, U_n\}$ is created. Then, the central node notifies the n users that they belong to the same group. The users receive a message with the IP addresses and the ports of the other members of the group in order to establish a communication channel with them. After this step, users can send messages directly to each other and the central node is no longer needed.

5.1.1 Group Key Generation

1. Users $\{U_1, \ldots, U_n\}$ agree on a large prime p where $p = 2q + 1$ and q is a prime too. Next, they pick an element $g \in \mathbb{Z}_q^*$ of order q.
2. In order to generate the group key, each user U_i performs the following steps:
   (a) Generates a random number $a_i \in \mathbb{Z}_q^*$.
   (b) Calculates her own share $y_i = g^{a_i} \text{mod} p$.
   (c) Broadcasts her share y_i and receives the other shares y_j for $j = (1, \ldots, n)$, $j \neq i$.
   (d) Uses the received shares to calculate the group key: $y = \prod_{1 \leq j \leq n} y_j = g^{a_1} \cdot g^{a_2} \cdot \ldots \cdot g^{a_n}$

5.1.2 Anonymous Query Retrieval

1. User U_i encrypts her query m_i:
   (a) U_i generates a random number r_i.
   (b) U_i encrypts her query m_i with the group key y: $c_i^0 = E_y(m_i, r_i) = (g^{r_i}, m_i \cdot y^{r_i}) = (c1_i, c2_i)$
2. For $i = (2, \ldots, n)$, each user U_i sends c_i^0 to the first member of the group (U_1).
3. For $i = (1, \ldots, n-1)$, each user U_i performs the following operations:
   (a) Receives the list of ciphertexts $\left\{c_1^{i-1}, \ldots, c_n^{i-1}\right\}$.
   (b) Using her share of the group key, partially decrypts the list of ciphertexts using the algorithm described in Sect. 4.1. The resulting list of ciphertexts is denoted as $\left\{c_1^{i-1'}, \ldots, c_n^{i-1'}\right\}$.
   (c) The list of ciphertexts $\left\{c_1^{i-1'}, \ldots, c_n^{i-1'}\right\}$ is re-masked using the re-masking algorithm described in Sect. 4.2 with a key $y' = \prod_{j=i+1}^{n} g^{\alpha_j}$. As a result, U_i obtains a re-encrypted version $\left\{e_1^{i-1}, \ldots, e_n^{i-1}\right\}$.
   (d) Permutes the order of the ciphertexts at random, obtaining a reordered version $\left\{e_{\sigma(1)}^{i-1}, \ldots, e_{\sigma(n)}^{i-1}\right\}$
   (e) Sends the list of ciphertexts $\left\{c_1^i, \ldots, c_n^i\right\} = \left\{e_{\sigma(1)}^{i-1}, \ldots, e_{\sigma(n)}^{i-1}\right\}$ to U_{i+1}.
4. The last user U_n performs the following operations:
   (a) Receives the list of ciphertexts $\left\{c_1^{i-1}, \ldots, c_n^{i-1}\right\}$.
   (b) Using her share of the group key, partially decrypts the list of ciphertexts using the algorithm described in Sect. 4.1 again. At this point, U_n owns the cleartexts of the queries.
   (c) Broadcasts the queries to the rest of users $\{U_1, \ldots, U_{n-1}\}$.

5.1.3 Query Submission and Retrieval

1. Each group member U_i submits the n received queries to the WSE.
2. Each user only takes the answer that corresponds to her original query.

5.2 Privacy Analysis

In [26], a complete privacy analysis of the protocol is provided. However, we next present a summary of this analysis in presence of the three possibly dishonest entities that participate in the protocol: a user, the central node, and the WSE.

- *Dishonest user* Similarly to [17], in order to guarantee the correctness of the process, the protocol assumes a scenario where users follow the protocol and there are no collusions.
- *Dishonest central node* This entity only participates in the initial phase of the protocol, ignoring any further communication between the users. Therefore, the central node cannot link any query to its source and hence, it is not a threat for the privacy of the users.
- *Dishonest web search engine* The objective of the WSE is to gather the queries of the users in order to build their profiles. However, when the proposed protocol is executed, the WSE cannot know whether a certain query has been generated by the user who has submitted it. This happens because when a user U executes the protocol, she submits her query hidden among other $n-1$ queries. Therefore, the WSE can correctly select the query that belongs to U with a probability of $\frac{1}{n}$. Note that, in order to build a useful profile, it is not enough for the WSE to correctly select one query in one execution of the protocol. The probability of correctly linking several queries to a user during a long period decreases exponentially.

5.3 Performance Analysis

In order to evaluate the performance, the proposed protocol was implemented and tested in a real environment. All these tests and the results obtained are detailed in [26], and they show that the query delay obtained by this protocol outperforms all the other proposals. More specifically, results indicate that the average query delay achieved by the protocol is 3.2 seconds.

According to [17], the UUP protocol obtained a query delay of 5.2 s [17]. However, the size of the results page returned by Google increased significantly in the period of time between the simulations of [17, 26]. This affects the total query delay. For this reason, in [26] it is stated that, in the same conditions, the UUP protocol obtains

a query delay of 6.8 s when the protocol of [26] obtains a query delay of 3.2 s. Regarding other proposals, it also ouperforms the fastest multi-party protocol with dynamic groups: [13] achieves a query delay of 3.9 s in a simulated scenario.

6 Contribution-2: Providing Privacy Against Dishonest Internal Users

This section presents a multi-party protocol that protects the privacy of users against web search engines and dishonest internal users. Regarding similar approaches, the protocol increases the level of security of [17], and requires less computation and communication than [18]. This work was published in [5, 27].

6.1 Protocol Description

The protocol is composed of four phases that are executed sequentially by the users.

Group setup Every user who wants to submit a query to the WSE, contacts the central node. When the central node has received n requests, it creates a group $\{U_1, \ldots, U_n\}$. Then, the n users are notified that they belong to the same group. The users receive a message with the size of the group (n) and the position that every component has been randomly assigned ($i = 1, \ldots, n$). Each position is associated with the IP address and the port where the user is listening. This information allows the users to establish a communication channel between them. The central node is no longer needed.

Permutation network distribution As stated in Sect. 4.3, t OAS-Benes networks are necessary in order to perform a secure permutation. The number of inputs of the networks equals the number of users $N = n$, which is also the same as the number of queries. Regarding the number of honest users, the parameter is always fixed at $\lambda = 2$ (see [5, 27] for further details).

Through knowing the parameters n, N, and λ, users calculate the value of t using the formula defined in Sect. 4.3. The construction of the t OAS-Benes $PN^{(n)}$ is mechanical. This means that users do not need to exchange any information. As long as they know the parameters t and n, they know the arrangement of the switches in the t OAS-Benes $PN^{(n)}$. Therefore, they can fairly divide them into n adjacent stages.

According to the positions assigned in the previous phase, user U_i takes responsability for the switches that correspond to the i-th stage. Each stage is formed by d switches, where $d = \frac{t}{n} \cdot S(n)$ on average.

We denote as s_l^i the l-th switch of the i-th user for $i = 1, \ldots, n$ and $l = 1, \ldots, d$. We also define a function $\Phi\,(i, l)$ that, given an output of a switch, it returns the input

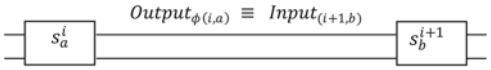


Fig. 5 Correlation between the outputs of a switch and the inputs of the next switch

of the next switch that it must follow. The result is given according to the arrangement of the switches in the PNs. Figure 5 illustrates the operation of this function.

6.1.1 Group Key Generation

The generation of the group key includes the following steps:

1. Users $\{U_1, \ldots, U_n\}$ agree on a large prime p, where $p = 2q + 1$ and q is a prime too. Next, they pick an element $g \in \mathbb{Z}_q^*$ of order q.
2. In order to generate the group key, each user U_i performs the following steps:
   (a) Generates a random number $a_i \in \mathbb{Z}_q^*$.
   (b) Calculates her own share $y_i = g^{a_i} \text{mod} p$.
   (c) Broadcasts a commitment to her share $h_i = \mathcal{H}(y_i)$, where $\mathcal{H}$ is a one-way function.
   (d) Broadcasts y_i to the other members of the group.
   (e) Checks that $h_j = \mathcal{H}(y_j)$ for $j = (1, \ldots, n)$.
   (f) Calculates the group key using the received shares: $y = \prod_{1 \leq j \leq n} y_j = g^{a_1} \cdot g^{a_2} \cdot \ldots \cdot g^{a_n}$

6.1.2 Anonymous Query Retrieval

For $i = 1, \ldots, n$, each user U_i performs the following operations:

1. U_i generates a random value r_i and uses the group key y to encrypt her query m_i:

$$E_y(m_i, r_i) = (c1_i, c2_i) = c_i^0$$

2. U_i sends c_i^0 to the other members U_j, for $\forall j \neq i$.
3. For every switch s_l^i ($l = (1, \ldots, d)$) with two inputs denoted as c_{i-1}^{2l-1} and c_{i-1}^{2l} received from U_{i-1} (note that the inputs for the switches of U_1 are the initial ciphertexts $\{c_1^0, \ldots, c_n^0\}$):
   (a) U_i re-masks the cryptograms c_{i-1}^{2l-1} and c_{i-1}^{2l}. She obtains a re-encrypted version e_{i-1}^{2l-1} and e_{i-1}^{2l}, using the re-masking algorithm defined in Sect. 4.2.
   (b) U_i randomly chooses $b \in \{0, 1\}$ to determine the state of the switch s_l^i as in Fig. 1a. According to this state, she obtains a re-ordered version of the ciphertexts $e_{i-1}^{\pi(2l-1)}$ and $e_{i-1}^{\pi(2l)}$.

(c) U_i broadcasts $\{c_{\Phi(i,2l-1)}, c_{\Phi(i,2l)}\} = \{e_{i-1}^{\pi(2l-1)}, e_{i-1}^{\pi(2l)}\}$

(d) Assuming:

$$c_{i-1}^{2l-1} = E_y(m_1, r_1), c_{i-1}^{2l} = E_y(m_2, r_2)$$
$$e_{i-1}^{\pi(2l-1)} = E_y(m_1', r_1'), e_{i-1}^{\pi(2l)} = E_y(m_2', r_2')$$

U_i must demonstrate that $e_{i-1}^{\pi(2l-1)}$ and $e_{i-1}^{\pi(2l)}$ are re-masked and re-ordered versions of c_{i-1}^{2l-1} and c_{i-1}^{2l}. This is equivalent to proving the two following statements:

I. $(m_2 = m_2') \vee (m_2 = m_1')$.
This can be proved by using the DISPEP protocol of Sect. 4.5.

II. $m_1 \cdot m_2 = m_1' \cdot m_2'$.
U_i computes $c = E_y(m_1 \cdot m_2, r_1 + r_2)$ and $c' = E_y(m_1' \cdot m_2', r_1' + r_2')$, and uses the PEP protocol (Sect. 4.4) to prove that c and c' are plaintext equivalent.

All the other users U_j $(\forall j \neq i)$ perform verification on the proofs.

4. Let us denote $\{c_1, \ldots, c_n\}$ the resulting list of re-masked and re-ordered ciphertexts. At this point, each user U_i owns those n values. Then, user U_i decrypts the value c_i that corresponds to a query m^i generated by one of the group members. Note that due to the re-masking and permutation steps, m^i probably does not correspond to m_i (the query that has been generated by U_i).

Decryption of a certain c_i requires that all n users participate by sending their corresponding shares to user U_i. According to that, U_i receives $(c1_i)^{\alpha_j}$ from U_j, for $j = (1, \ldots, n)$ and $j \neq i$. Then, U_i computes her own share $(c1_i)^{\alpha_i}$. Finally, U_i retrieves m^i by computing:

$$m^i = \frac{c2_i}{c1_i^{\alpha_i}(\prod_{j \neq i} c1_i^{\alpha_j})}$$

6.2 Privacy Analysis

Similarly to Sect. 5.2, this section presents a summary of privacy analysis detailed in [5, 27], evaluating privacy in front of the same three adversaries:

- *Dishonest user* Every time that a ciphertext c_i crosses a switch, it is re-masked and permutated, and the attacker can only link the result to c_i by random guessing, with a probability of success of $1/2$. This probability exponentially decreases for every switch the ciphertext crosses.

  In the case of an attacker that only knows the inputs and the final outputs of the protocol, the intermediate re-maskings and permutations prevent her from finding the links between them. Hence, given a particular user, the probability of correctly linking her with a decrypted query is $1/n$.

Let us consider the case where a dishonest user successfully learns the query of another component of the group. This means that she is able to link one input of the permutation networks with one of the outputs. This attack may be conducted if one of the following conditions is fulfilled.

1. *The dishonest user knows the secret group key*. In this case, the attacker can decrypt the queries at any step of the protocol.
2. *The dishonest user ignores the key but knows the overall permutation*. In this case, the attacker waits until the ciphertexts are decrypted. Then, she can link every query with the original ciphertexts and, hence, with their sources.

Regarding the first condition, the attacker can only recover the secret key if she compromises the $n-1$ other members of the group. The generation of the group key is distributed among the participants using the n-out-of-n threshold ElGamal key generation explained in Sect. 4.1. One of the characteristics of this scheme is that, even if there is a single honest user, the secret key cannot be reconstructed.

Another alternative in order to learn the secret key is to maliciously alter the key generation phase. In this phase, each user generates her share $y_i = g^{a_i}$, then, she broadcasts a commitment to that share using a cryptographic function $\mathcal{H}(y_i)$, and then she sends y_i in a new message. A dishonest user may change her choice of share after receiving the shares of the other participants, before sending her own. This dishonest user calculates her share $y'_j = g^{a_j} / \prod_{i=1}^{n-1} y_i = g^{a_j - a_1 - \cdots - a_{n-1}}$ and broadcasts it. As a result, the group key is computed as $y = g^{a_j}$ and, hence, the dishonest user knows the secret group key.

In order for this attack to be successful and to remain undetected, the dishonest user must be able to find collisions in the hash function. This means that she must find a value y'_j for which her previous commitment is still valid (i.e., $\mathcal{H}(y_i) = \mathcal{H}(y'_i)$). Nowadays, the probability of finding a collision in a reasonable amount of time, using a cryptographic hash function such as SHA-2, is almost negligible.

Regarding the second condition, the use of OAS-Benes PNs guarantees that the permutation remains random and private. The requirement that must be satisfied is that there must be at least one permutation network controlled by honest users. This means that the proposed scheme needs a quantity of PNs that depend on the minimum number of honest users required to run the protocol. More specifically, the quantity of PNs that the scheme needs is the number that satisfies the following condition: in any possible distribution of stages among the users, the amount of switches controlled by the t honest users equals, at least, the number of switches composing one OAS-Benes PN. If this requirement is fulfilled, according to [23], the permutation is secure and remains secret to all the participants. Then, it is not possible to backtrace a permutation to find the original input.

- *Dishonest central node* As in Sect. 5.2, the central node only participates in the initial phase of the protocol, and cannot link any query to the source.
- *Dishonest web search engine* The WSE can link each query with the user who submitted it and include that information in her profile. Since a user U_i does not

always submit her own query but the query of another participant, her profile is distorted. Hence, after several executions of the protocol, the profile of U_i owned by the WSE is obfuscated and her privacy is protected.

6.3 Performance Analysis

The performance analysis described in [5, 27] includes a comparison of the system with similar proposals. Systems are analyzed in terms of computation (number of modular exponentiations required by each protocol), and in terms of network usage (number of exchanged messages in each protocol).

Results show that, for the only protocol which achieves a similar level of privacy ([18]): (i) regarding the modular exponentiations, [18] requires a higher computation time. For example, for a group $n = 3$ users, it needs 2 more seconds than the protocol presented in Sect. 6.1; and (ii) although the number of messages is similar in both proposals, the protocol presented in Sect. 6.1 requires less message deliveries than [18].

7 Conclusions

Web search engines play an important role in the use of the Internet. However, while users benefit from these services, their privacy may be seriously threatened. For this reason, several proposals in the literature have appeared to address this problem.

This work has presented an analysis of the current multi-party proposals that provide privacy to users of web search engines. This analysis shows that there is a trade-off between the level of security that private web search tools can provide, and the response time that the users may experience.

Our contributions to private web search consist in presenting two proposals that maximize one of these features (i.e., the query delay or the level of security). Both proposals are based on the idea of distorting a profile by preventing the web search engine from knowing the real source of a submitted query. Our work is built upon an analysis of an existing multi-party protocol, the UUP protocol [17].

The first proposal considers a scenario where users need to receive the response for their query as fast as possible. Hence, the proposed protocol focuses on obtaining a low query delay. The scheme was tested in an open environment and the results show that it achieves the lowest query delay that has been reported in the literature. This work was published in [26].

The second proposal focuses on enhancing the level of security. The scenario considers the presence of dishonest internal users. The privacy analysis shows that

users are protected against the web search engine and against dishonest internal users. Regarding performance, it ouperforms similar proposals with the same level of privacy. This proposal was published in [5, 27].

Acknowledgments Authors are solely responsible for the views expressed in this text, which do not necessarily reflect the position of UNESCO nor commit that organization. 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, Audit Transparency Voting Process IPT-430000-2010-31, ICWT TIN2012-32757 and BallotNext IPT-2012-0603-430000) and by the Government of Catalonia (under grant 2009 SGR 1135).

References

1. Barbaro, M., Zeller, T.: A face is exposed for aol searcher no. 4417749. New York Times (2005)
2. Hafner, K., Richtel, M.: Google resists u.s. subpoena of search data. New York Times (2006)
3. Cooper, A.: A survey of query log privacy-enhancing techniques from a policy perspective. ACM Trans. Web. **2**, 19:1–19:27 (2008)
4. Viejo, A., Castellà-Roca, J., Bernado, O., Mateo-Sanz, J.M.: Single-party private web search. In: Proceedings of the 2012 Tenth Annual International Conference on Privacy, Security and Trust (PST). PST '12, Washington, DC, USA, IEEE Computer Society, pp. 1–8 (2012)
5. Romero-Tris, C., Castellà-Roca, J., Viejo, A.: Multi-party private web search with untrusted partners. In: 7th International ICST Conference on Security and Privacy in Communication Networks -SecureComm'11 (2011)
6. Domingo-Ferrer, J., Solanas, A., Castellà-Roca, J.: h(k)-private information retrieval from privacy-uncooperative queryable databases. Online Inf. Rev. **33**, 720–744 (2009)
7. TrackMeNot: TMN. http://mrl.nyu.edu/dhowe/trackmenot (2013)
8. Murugesan, M., Clifton, C.: Providing privacy through plausibly deniable search. In: SDM (2009)
9. Sánchez, D., Castellà-Roca, J., Viejo, A.: Knowledge-based scheme to create privacy-preserving but semantically-related queries for web search engines. Inf. Sci. **218**, 17–30 (2013)
10. Arampatzis, A., Efraimidis, P., Drosatos, G.: A query scrambler for search privacy on the internet. Inf. Retr. **16**:6, 657–679 (2013)
11. Viejo, A., Sánchez, D.: Providing useful and private web search by means of social network profiling. In: Proceedings of the 2013 Eleventh Annual International Conference on Privacy, Security and Trust (PST). PST '13, To appear (2013)
12. Reiter, M., Rubin, A.: Crowds: anonymity for web transactions. ACM Trans. Inf. Syst. Secur. **1**, 66–92 (1998)
13. Viejo, A., Castellà-Roca, J.: Using social networks to distort users' profiles generated by web search engines. Comput. Netw. **54**, 1343–1357 (2010)
14. Wright, M.K., Adler, M., Levine, B.N., Shields, C.: The predecessor attack: an analysis of a threat to anonymous communications systems. ACM Trans. Inf. Syst. Secur. **7**, 489–522 (2004)
15. Erola, A., Castellà-Roca, J., Viejo, A., Mateo-Sanz, J.M.: Exploiting social networks to provide privacy in personalized web search. J. Syst. Softw. **84**, 1734–1745 (2011)
16. Domingo-Ferrer, J., Bras-Amorós, M., Wu, Q., Manjón, J.: User-private information retrieval based on a peer-to-peer community. Data Knowl. Eng. **68**, 1237–1252 (2009)
17. Castellà-Roca, J., Viejo, A., Herrera-Joancomarti, J.: Preserving user's privacy in web search engines. Comput. Commun. **32**, 1541–1551 (2009)
18. Lindell, Y., Waisbard, E.: Private web search with malicious adversaries. In: Proceedings of the 10th International Conference on Privacy Enhancing Technologies—PETS'10, pp. 220–235 (2010)

19. Desmedt, Y., Frankel, Y.: Threshold cryptosystems. In: Computer Science, L.N. (ed.) Advances in Cryptology—CRYPTO'89, vol. 335, pp. 307–315 (1990)
20. ElGamal, T.: A public-key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inf. Theory **31**, 469–472 (1985)
21. Chaum, D., Pedersen, T.: Wallet databases with observers. In: Computer Science, L.N. (ed.) Advances in Cryptology—CRYPTO'92, vol. 740, pp. 89–105 (1992)
22. Abe, M.: Mix-networks on permutation networks. In: Computer Science, L.N. (ed.) Advances in Cryptology—Asiacrypt'99, vol. 1716, pp. 258–273 (1999)
23. Soo, W.H., Samsudin, A., Goh, A.: Efficient mental card shuffling via optimised arbitrary-sized benes permutation network. In: Computer Science, L.N. (ed.) Proceedings of the 5th International Conference—ISC 2002, vol. 2433, pp. 446–458 (2002)
24. Jakobsson, M., Juels, A.: Millimix: mixing in small batches. DIMACS Technical report 99–33 (1999)
25. Schnorr, C.P.: Efficient signature generation by smart cards. J. Cryptol. **4**, 161–174 (1991)
26. Romero-Tris, C., Viejo, A., Castellà-Roca, J.: Improving query delay in private web search. In: 3PGCIC, pp. 200–206 (2011)
27. Romero-Tris, C., Castellà-Roca, J., Viejo, A.: Distributed system for private web search with untrusted partners. Comput. Netw. **67**, 26–42 (2014)

DisPA: An Intelligent Agent for Private Web Search

Marc Juarez and Vicenç Torra

Abstract Search queries can be used to infer preferences and interests of users. While search engines use this information for, among others, targeted advertising and personalization, these tasks can violate user's privacy. In 2006, after AOL disclosed the search queries of 650,000 users and some of them were re-identified, many Privacy Enhancement Technologies (PETs) have sought to solve this problem. The *Dissociating Privacy Agent* (DisPA), is a browser extension that acts as a proxy between the user and the search engine and semantically dissociates queries on real time. We show that DisPA increases the privacy of the user and hinders re-identification. We also propose an algorithm to measure and evaluate the privacy properties offered by DisPA.

1 Introduction

Web search has become an elementary task in the Internet and virtually everybody makes use of search engines to find information in a quick and effective way. Providers of search services log data related to their users and track them across the Web. The reason is twofold. Firstly, profiling is profitable for the provider who can exploit business opportunities by means of Marketing Research and Targeted Advertising [1]. Secondly, due to the growth of the Web, profiling is essential to improve ranking algorithms and offer a more efficient search [2–4].

The task of profiling consists in the collection of data about the user's web interaction. These data are stored in files at the server-side called "server logs" or "query logs". By applying data mining techniques on these logs, search providers extract

M. Juarez (✉)
KU Leuven, Department of Electrical Engineering (ESAT), COSIC, iMinds, Kasteelpark Arenberg, 3000 Leuven, Belgium
e-mail: marc.juarez@esat.kuleuven.be

V. Torra
Institut d'Investigació en Intel·ligència Artificial, Consejo Superior de Investigaciones Científicas Campus de la UAB, 08193 Bellaterra, Catalonia, Spain
e-mail: vtorra@iiia.csic.es; vtorra@ieee.org

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_21

traits of the user such as demographic aspects (e.g., age, gender or nationality) or main areas of interest, that are modelled as categories such as "Cinema" or "Football". Afterwards, the ranking algorithms rearrange the list of results to deliver first those that are more useful for the user according to his preferences [5].

This personalization can be beneficial because saves time to the user who otherwise would have to skim over the large list of results manually. Nevertheless, the indiscriminate logging of data raises privacy concerns with respect to social sorting and discrimination. As it has been shown several times in the past, potentially sensitive information can be inferred from search queries, such as sexual orientation, health status, or political beliefs [6].

A milestone in history of privacy breaches is the AOL search data leak in 2006 [7], when queries of approximately 650,000 users submitted over a 3-month period were disclosed [8]. Despite that AOL claimed to have anonymized the dataset by removing identifiers, journalists of the New York Times managed to link one of the logs to a real identity [9]. This was very remarkable as it proved that queries by themselves can uniquely identify an individual or, at least, reduce the search space considerably.

Several approaches are commonly taken to address this problem. First, the user can use cryptography-based solutions which provide strong privacy guarantees, but require the provider to integrate them in the backend [10]. Second, the user can connect to the service through an anonymous communication system that would provide him a different identity for each session. Finally, he might still be identifiable and obfuscate, either the content of the queries, or his search profile by means of dummy queries [11].

In this paper we describe a Privacy Enhancing Technology (PET) for web search that has been developed through the last two years [12, 13]. It tackles the problem from still another perspective that is characterized by taking into account search personalization. We assume that the user benefits from personalization and, for this reason, we strive for a trade-off between the utility (personalization) and the cost (privacy) of releasing data.

The rest of the paper is organized as follows. Section 2 reviews the state-of-the-art in private web search. In Sect. 3 we present our threat model and recall the basic operation of the approach we propose: the *Dissociating Privacy Agent* (DisPA for short) [12]. In Sect. 4 we detail the internal operations of DisPA. In Sect. 5 we present the different experiments conducted for the evaluation of the agent and show the results obtained. In Sect. 6 we point out the limitations of our work and bring some discussion points. The paper finishes with the conclusions and lines of future research in Sect. 7.

2 Related work

There exist several cryptography-based solutions for private search: private information retrieval (PIR) [14, 15], oblivious transfer (OT) protocols [16], and methods based on homomorphic cryptosystems (e.g., Paillier) [17]. These protocols provide

strong privacy guarantees such as confidentiality of search terms and results. However, there are some drawbacks for their implementation in a real-world web search engine. For instance, they often assume cooperation by the provider. Search providers however do not have any incentives to implement costly protocols they cannot profit from, and thus the deployment of these solutions is not realistic in practice. Further, given that search terms are encrypted, they render useless for personalization. Another important difficulty that makes them inconvenient is the computational complexity of these methods given the great size of the Web.

For these reasons, the problem is often relaxed towards more applicable schemes. Among these, we find two main different strategies: (i) to obfuscate the user's profile by submitting fake queries together with legitimate ones, and (ii) to hide the identity of the user in front of the search engine, so queries cannot be attributed to him. The former category is often called *obfuscation-based* techniques and it has been thoroughly studied [18–26]. We refer the reader to a deeper analysis of obfuscation techniques for more details [27].

Most of low-latency anonymous communications systems fall in the latter category. For instance, the user could employ The Onion Router (Tor) [28] to submit the query. The server would observe the IP of a different Tor exit node for each session, making it harder to link user's queries across sessions. This approach has two important limitations. First, as the AOL case demonstrated, queries by themselves can identify users independently of the communication's metadata. Second, due to the time overhead introduced by Tor, it deteriorates the usability of the service and therefore it is not suitable for a long-term solution.

Besides of the particular shortcomings that each of these approaches have, they have a drawback in common: all of them diminish the quality of server logs for profiling. Obfuscation-based techniques introduce false information about the preferences of the user, and anonymity networks induce the creation of a new server log for each session.

Our research fills this gap by proposing an intelligent agent that helps to protect the user's privacy, while preserving the utility of his profile for personalization. There are different strategies to achieve this goal in the literature [20, 29–31], and we also have found in recent publications very similar approaches to the ones we had presented for the development of DisPA [32, 33]. This shows a common interest of the research community towards the development of protocols that strive for a trade-off between utility and privacy in web search.

3 Threat Model and Fundamentals

We assume that the adversary is the search engine provider or a third party who has access to all server logs. The goal of the adversary is to extract new information about a targeted user out of the logs or, in the worse case from the user's point of view, to discover the real identity of the user.

We can model the adversary as a honest-but-curious adversary. This means a passive adversary who does not alter the functionality of the system but can eavesdrop queries and analyse them. Search providers fit in this model because they are interested in ensuring the availability and good quality of the service.

The adversary might as well have some auxiliary knowledge that enables re-identification. As it has been shown in the past, an adversary can use cross-correlation with multiple databases to uniquely identify an individual (e.g., Narayanan and Shmatikov showed it with popular movie databases [34]). We will discuss in more detail this problem in Sect. 6.

Before diving into the fundamentals of the agent we are going to describe our system model. A query is basically an HTTP(S) request from the user's browser to the search engine's web server. The URL field contains query terms and other user preferences encoded as URL parameters. The cookie field contains, among other domain-specific cookies, a cookie with a user's unique identifier, the so-called "cookie ID". The query terms are stored in a server log along with other connection-related information, such as the IP and the cookie ID.

We make the assumption that search engines only use cookies to identify users. This might be a strong assumption to hold, but the last version of Google's privacy policy and a recent study on log retention policies support it [35, 36]. Also, a recent study on the prevalence of "device fingerprinting", a new tracking technique that leverages information collected about devices for user identification, has not found evidence of the adoption of such technique in most popular search engines [37].

3.1 The Dissociating Privacy Agent

The underlying concept of our approach is "dissociation". Dissociation is based on the observation that users are multifaceted individuals, in the sense they are interested in various areas of knowledge, such as "Science", "Sports" or "Music". Let $C(u) = \{c_1, c_2, \ldots, c_n\} \subset \mathcal{C}$ be the set of categories of interest for a particular user u. Let U_c be the set of users interested in category c. Then, the anonymity set of u is defined as

$$A(u) := U_{c_1} \cap U_{c_2} \cap \cdots \cap U_{c_n}.$$

Our hypothesis is that, for a fine-grained taxonomy $\mathcal{C}$, $A(u)$ is likely to contain only user u. Put differently, the interests of the user define his identity and can be used to uniquely identify him.

The idea of dissociation is to break down the identity of the user into partial identities, each one of them grasping a fraction of his interests. We name these artificial identities as "virtual identities". If we consider each virtual identity as a different user, dissociation increases the probability of users sharing the same interests. It is trivial to prove that $|A(u)| \leq |A(u_i)|$, where u_i are the virtual identities of u after dissociation, for $i = 1, \ldots, n$.

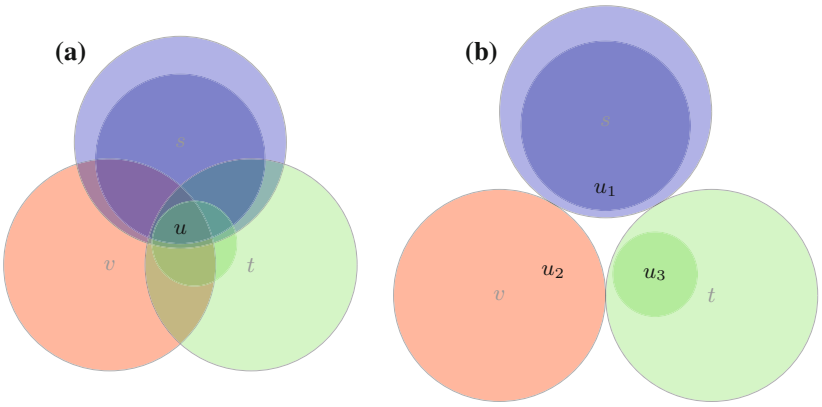


Fig. 1 Venn diagrams showing the dissociation process for user u

To illustrate this we show in Fig. 1 an example of dissociation. Each circle represents a set U_{c_i} and each user is represented with a letter: s, t, u, v. In Fig. 1a we see that the anonymity set of u only contains u. We apply dissociation on u by creating a virtual identity for each of the set U_{c_i} for the three main facets: u_1, u_2, u_3. As a result, in Fig. 1b we see how the new anonymity sets $A(u_i)$ have size 2.

4 Design

In order to implement dissociation, we designed the Dissociating Privacy Agent (DisPA). DisPA is an intelligent agent, i.e., a piece of software that takes decisions on behalf of the user. The agent is implemented as a browser add-on and acts as a proxy between the browser and the web server.

To achieve dissociation, DisPA intercepts HTTP requests to the search engine. Then, the connection is bypassed through a query classifier. DisPA generates new tracking data for each possible classification outcome and replaces them in the HTTP request on real-time. To the eyes of the server, queries classified by DisPA in different categories appear as requests from different users and thus are logged into different files at the server-side (see Fig. 2).

To keep consistency across different HTTP sessions, we define a context in the browser formed by: the jar of cookies, history of queries, history of clicked links, lists of results and the user-agent. This is intended to prevent the server from spotting similarities among sessions and link them to the same user.

As a result, the identity of the user is divided and it is harder to achieve re-identification by means of dissociated logs. Furthermore, note that dissociated logs are still useful for profiling as, by construction, they preserve partial but real user interests that search engines can extract and use for personalization.

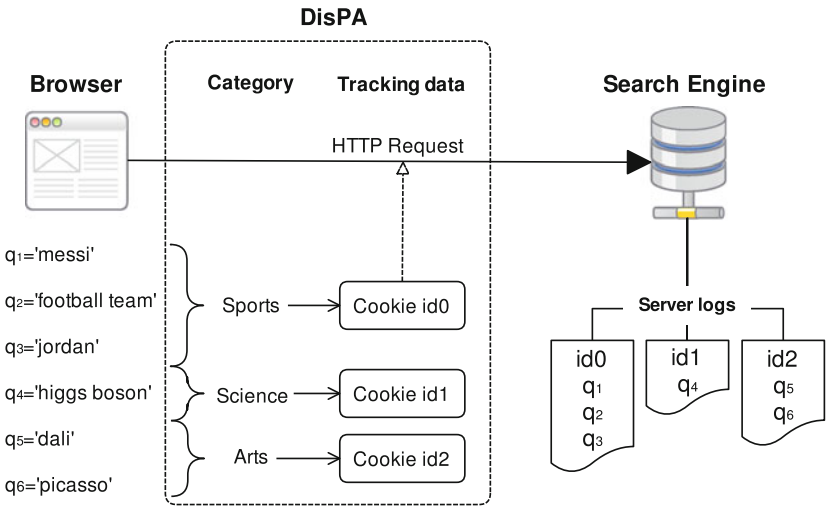


Fig. 2 Representation of the implementation of dissociation in DisPA

4.1 Query Classification

A fundamental part of the agent is query classification. The user sends queries through the browser add-on's interface and the agent classifies them before submitting them to the search engine. DisPA uses the taxonomy of the Open Directory Project (ODP) to build a faceted search engine and classify queries quickly [38].

The facets of the user are modelled as categories of the **first level** of the ODP tree which are:

$$Adult, Arts, Games, Shopping, Business, Health, Society, Computers, Home, News, Reference, Recreation, Sports, Science, Society.$$

In order to classify a query, we perform a faceted search in a local search engine, which is based on an inverse index of the documents in the ODP corpus. The outcome of the search is a vector with coordinates the number of hits of the query in each of the categories. That is, let q be a query, given the set of categories $C = \{c_1, c_2, \ldots, c_n\}$, the outcome of classification is a vector

$$(h_1, h_2, \ldots, h_n) \tag{1}$$

where h_i is the number of documents indexed in category c_i that are hit by query q, for $i = 1, \ldots, n$.

Given a query q_0, a very basic classifier can be defined as:

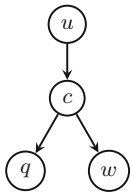


Fig. 3 Graphical model representation for PQC

$$classify(q_0) := \arg\max_{c_i \in C} p(c_i \mid q_0), \tag{2}$$

where in our approach we estimate $p(c_i \mid q_0)$ by

$$p\,(c_i \mid q_0) \approx \frac{h_i(q_0)}{\sum_i h_i(q_0)}.$$

4.2 Personalized Query Classification

Frequently, queries are vague because they lack of context and polysemy of words introduce ambiguity in their interpretation. For example, the query "jordan" might allude to a basketball player, a mathematician, a river, or a country. Personalized Query Classification (PQC) is more challenging than plain QC in that it attempts to resolve ambiguity of queries according to subjective user intents.

In order to take advantage of search engine's personalization, the dissociation process must be consistent with the user's interests. In other words, we need to perform PQC so that a user mainly interested in basketball gets the query "jordan" in the basketball profile and obtains results related to "Michael Jordan".

The probabilistic model used to achieve PQC in DisPA is similar to the one presented by Cao et. al. [39]. We contributed with a novel approach based on inferring user's interests by means of the history of navigation instead of the clicking data. The idea is that DisPA, as a browser add-on can take advantage of its direct access to the browser profile.

Our model can be described in terms of a user u who wants to search for information related to query q. The search engine interprets q as belonging to a set of categories. Independently of this, the user accesses web pages w that might be classified in this set of categories. Thus, w depends on c. In Fig. 3 we can see the graphical representation of our model.

The main takeaway is that we keep the classification outcome for the last k web pages to estimate prior probabilities of the user being interested in each category—i.e., $p(u \mid c) \approx p(w_1, \ldots, w_k \mid c)$. Then, these priors are used to weight the final classification for a specific query q.

We skip the mathematical details and refer the interested reader to the original paper for a full specification [12]. The final classifier is

$$classify(q_0) = \arg\max_{c_i \in C} p(c_i \mid q_0)\, p(w_1, \ldots, w_k \mid c_i). \tag{3}$$

And we estimate $p(w_1, \ldots, w_k \mid c_i) \approx \frac{v_i}{|c_i|+v_i}$, where v_i is the number of visited sites classified in the i-th category of the ODP.

4.3 Filters of Queries

Our approach stems from the assumption that the user's identity is defined by his interests. DisPA enlarges anonymity sets and reduces the adversary's inference ability by dissociation. However, it is obvious that this is not sufficient to solve the problem. There are other types of queries that jeopardize user's privacy, for example:

1. Queries that identify the user by their own (e.g., a query containing unique identifiers or emails
2. Queries that contain named entities like names of locations or personal names (e.g., terms like "lilburn" and "arnold")

We characterize these types of queries as the most uncommon queries amongst the world of users in the search engine, as these are the ones that reveal more information. This fits in DisPA's model as people that are fond on rare topics are easier to identify. Even though their profiles are dissociated, the anonymity sets may remain very small because there are no other users interested in the same topics.

We approximate the popularity of a query by the number of results that our ODP-based classifier returns, that is the absolute frequency of the query in the ODP. Following the notation in the previous sections this corresponds to

$$f(q) \approx \sum_i h_i(q).$$

We set a threshold τ for the frequency, which can be initialized by submitting a very uncommon query locally. For subsequent queries, we test $f(q) < \tau$, and filter them out in case it is true. We also used the Stanford's CoreNLP library for Named Entity Recognition (NERQ) to recognize locations and personal names.

In order to filter a query out, the agent generates a new virtual identity exclusively for that particular query. This way we isolate these queries from the rest of profiles and cannot be used to neither extract new information nor to link the other dissociated logs of the user.

4.4 User Specializations

One of the main flaws of our first implementation was that classification categories are fixed and did not take into account user's specializations. A user may have a lot of facets but he might be interested in ones relatively more than in others. For example, compare queries sent from the computer at the work place and the ones that are sent from home.

In some cases, interests of the user are specialized. Then, most of queries are classified in one of DisPA's categories and dissociation makes no difference. As an example to illustrate this, imagine a user very keen on computers. His queries fall mostly in the category "*Computers*" and the other categories are barely used. As a result, the dissociation by means of the first level of the ODP has no effect and the agent fails in its attempt to protect the user.

An improvement on this first implementation consists in breaking down these specialized categories to include more specific categories that describe user's interest more accurately [13]. In the example above, computers would be expanded with the children of its node in the ODP tree: *AI, Algorithms, Games, Hacking, Internet, etc.* This way queries are sparser and dissociation would be effective.

Nevertheless, note the trade-off between privacy and personalization in this process. Categories range from broad topics (upper levels of the tree), to very narrow (lower levels), to the edge case of considering each individual query as a category. The former provides better personalization because it yields more data to the server. On the other hand, the latter obstructs personalization but provides more privacy.

Besides, we have to consider long-term and short-term interests. A user specialization may be temporal and change with time. DisPA copes with that by self-adapting over time and rearranging the set of categories for classification automatically. For instance, in the example above, if the user suddenly becomes more interested in "*Music*", the system should roll-back to the previous state by retracting the old category "*Computers*" and, afterwards, expand "*Music*".

Our approach to achieve this, we normalize the vector defined in Expression 1 and consider it as the distribution of probabilities of n random variables X_i representing the event of the query q belonging to the category c_i, for $i = 1, \ldots, n$. Then we measure the dispersion of this distribution by the coefficient of variation defined by $c_v := \frac{\sigma}{\mu}$.

We set a threshold that indicates when the number of queries per category is unbalanced and, thus, we have to expand or retract a category of the tree.

4.5 Self-Adaptive Classification

Recall from Sect. 4.1 that $C = \{c_1, \ldots, c_n\}$ is the set of categories used for classification. Rearrangement of C take place when a deviation from the man is detected.

The expansion operation occurs when the deviation is positive. In that case we add all the children of the category to be expanded into C. Note that we do not remove the parent from C because otherwise we would lose a possible outcome of the classification. For instance, imagine a future query that hits documents contained only found in the parent. In case that the deviation is negative, the category has too few queries and it must be dropped. Thus, if this happens with all the categories that share the same parent category, we aggregate the children into the parent.

Once we have some categories expanded, in order to classify a query, we perform a level-wise classification. We begin at the first level of the tree and find the category of maximum weight. Then, if it has been expanded, we find the child with maximum weight and so on with the lower levels, until the category that minimizes the dispersion at the current level has not been expanded.

During the expansion, we generate a new virtual identity for each child and the virtual identity of the parent stays the same. This way, the log of the parent in the server may contain queries of other subcategories but it does not affect personalization. When retracting a category we simply use the virtual identity of the parent that we preserved in the expansion operation and, if we expand it again, we reuse the old virtual identities for the children.

As a result, we are able to adjust the level of sparseness of the logs in the server and, thereby, adjust the trade-off between privacy and personalization. We refer the reader to the algorithms implemented for this process [13].

5 Empirical Results

In order to test the agent and prove that the risk of re-identification is reduced we used the linkage algorithm described in [12]. This algorithm is supposed to be applied by the adversary on the server logs and link those belonging to the same user together.

5.1 Evaluation

The lack of public sets of queries makes difficult the evaluation of the degree of personalization achieved by the agent as well as the effect that the agent has on it. As a preliminary experiment we submitted a set of 803 queries through DisPA from an AOL user and we did not notice any difference with plain search. Then, we submitted a set of 2,743 queries of another AOL user several times and there was a difference on the order of two results (from first to second place in the list). Nevertheless, a complete analysis of personalization is out of the scope of this work, we center our evaluation in the disclosure risk.

We developed an attacking algorithm against our own agent. Such an algorithm could be used by the adversary to rebuild the original user's server log out of the partial logs generated by DisPA. The algorithm is based on the observation that there

are terms that are more common in the user's queries than in other users'. These terms do not have strong semantic meaning and are classified according to the rest of the terms in the query. Consequently, these terms are spread by the dissociation over all partial logs. The algorithm tries to exploit this by linking the logs that contain these terms.

As mentioned in Sect. 4.3, an instance of these terms are named entities. This takes inspiration from the AOL case. The NYT journalists identified Thelma Arnold because she was looking for venues in her town ("lilburn") and names of relatives ("arnold"). Then, they used a telephone directory to narrow the search up to 14 individuals [9].

To detect the terms, the adversary has to represent logs as vectors using a tf-idf scheme. The rationale is that tf-idf reflects the importance of a term in a log offset over its frequency in the collection of all logs. Then, the algorithm clusters the vectorial space using the DBSCAN algorithm with the cosine similarity as distance.

The linkage algorithm initializes with one or more logs known to belong to the user (auxiliary knowledge), that we call "seeds". At the end, clusters containing a seed are joined into one unique cluster that represents the original log.

To evaluate the user's *disclosure risk*, that is the risk that the original log is recovered by the adversary after dissociation, we measure the quality of the clustering provided by the linkage algorithm. To evaluate the clustering we measure the F1-Score of the binary classification defined by the property of a query being part of the final cluster or not.

The F1-Score combines precision and recall. Note that in our case, true positives are queries of the targeted user that fall in the cluster, false positives are queries of other users that fall in, true negatives are queries of other users that fall out and false negatives are queries of the target user that fall out. A higher F1-Score corresponds to a higher success rate of the adversary.

The DBSCAN clustering requires a parameter as an input that defines the neighbourhood of a data point. This parameter is not known a priori by the attacker. We consider the worse-case for the user and find the value that gives the best clustering through experimentation.

5.2 Experiments

We used the AOL released dataset for the experiments. The dataset contains an user ID, the terms of the query and the URLs of the results that were clicked. We performed five experiments described next.

- Experiment 1: We took a sample of logs of 20 different users and submitted their queries through the agent (with query filters disabled). Then, we applied the clustering algorithm to the resulting dissociated logs taking a random seed.
- Experiment 2: We added Arnold's log to the sample of logs. We chose Arnold's log because her log contains named entities like "Arnold" or "Lilburn" in queries

that fall in different categories. We repeated the first experiment under the same conditions to see if the clustering algorithm performed better. This time we took as seed the dissociated log corresponding to the class "Arts", one of the largest dissociated logs. The justification is that it is more likely that the attacker finds information related to the user in the largest log.

- Experiment 3: For the third experiment we repeated the second experiment but enabling the filter of uncommon queries and treating queries with named entities as described at Sect. 4.3.
- Experiment 4: This experiment is intended to evaluate the self-adaptive DisPA. For this experiment we did not use the AOL dataset because logs are very small to be specialized enough. Instead, we developed a generator of queries based on the keywords stored in the ODP that we referred in Sect. 3. The generator takes a probability distribution for the classification taxonomy as an argument and generates a log of queries accordingly. For this experiment we used the following distribution:

$$
\begin{aligned}
Adults &\ 0 \\
Arts &\ 0 \\
Games &\ 0.02 \\
Reference &\ 0.02 \\
Shopping &\ 0 \\
Business &\ 0.04 \\
Health &\ 0.02 \\
News &\ 0 \\
Society &\ 0.1 \\
Computers &\ 0.8 \\
Home &\ 0 \\
Science &\ 0 \\
Sports &\ 0
\end{aligned}
$$

  We simulated the submission of these queries first using first implementation of DisPA and, then, the self-adaptive version setting the threshold of the coefficient of variation to 80 %. We added 20 random users from the AOL released dataset and applied the linkage algorithm.
- Experiment 5: we repeated the fourth experiment but using the self-adaptive version of DisPA.

In order to claim whether the user is protected or not, we set 50% of disclosure risk as a threshold. If F1-Score is below this threshold we say that the user is protected, and not protected otherwise.

5.3 Results

For the first experiment we found that for small values of ε the algorithm reaches maximum precision because the final cluster only contains the seed. All queries in the seed log were queries that fell in the final cluster (true positives) and there were no queries of other users (false positives). In contrast, the recall is zero because all-but-one server logs of the user fell out of the final cluster (false negatives).

This translates to a low F1-Score and hence a low disclosure risk. As we increase ε more and more logs fell into the final cluster. Nevertheless, this server log was well dissociated by DisPA and the algorithm, for the given seed, jumped directly to the situation where the whole collection of server logs fell into the final cluster. This means that user's logs could not be linked using the algorithm with this seed because these logs did not have enough rare terms in common.

For the second experiment we used Thelma Arnold's log as the target log. We saw that for $\varepsilon = 1.39$ we had the optimal clustering. The algorithm had linked most of the dissociated logs and, thus, if offered a disclosure risk close to 90 %.

For the third experiment we took the same parameters for the attacking algorithm but this time we used the filter for uncommon queries described in Sect. 4.3. The disclosure risk was almost constant for the same interval of values taken in the previous experiments. For $\varepsilon = 1.9$ disclosure risk increased. This means that logs could not be linked because uncommon terms had been successfully separated in different logs.

In the fourth experiment, we showed there were some values of the neighbourhood distance for which the user was not protected because the F1-Score was above 50 %. We saw that it made no sense to go on evaluating greater values than 2 because the precision was maximum. This means that all targeted logs were falling in the final cluster and the clustering was not going to improve. In fact, we actually saw all logs in the server fell in the final cluster since recall was very low.

Finally, in the fifth experiment we did exactly the same, although the seeds changed because we were considering a different collection of dissociated logs. In fact, two categories were expanded during the simulation: "$Top/Computers$" and "$Top/Computers/Internet$".

For this last experiment the disclosure risk was below 50 % for all ε and, therefore, the user was protected. The percentage of disclosure risk reduction from the standard DisPA in the worst case was around 67 %.

6 Discussion and Limitations

One of the main limitations of this work is the assumption of search engines exceptionally using cookies to track users. This is a strong assumption to hold today, specially after the revelation of the increasing prevalence of device fingerprinting. However, there is no evidence yet of any search engine using these techniques at the

moment. As a very rough countermeasure the agent replaces the user-agent string by a more general one. We extracted this user-agent from a small-scale panopticlick-like survey that we conducted in our circles of acquaintances.[1]

We must admit that the adversary considered is not a fully strategic adversary. In the experiments we are facing a particular algorithm of re-identification and it might be that a manual inspection could further reveal other information that the algorithm misses. In favour of the algorithmic approach we must say that given the amount of logs in real-world servers, any manual approach would be infeasible.

A limitation is that dissociated profiles might deviate significantly from the average profile of the population of users. This effect can be detected by an adversary who can extract statistics from the server. However, we cannot prove this given that we do not have enough information about the distribution of profiles in real-world servers. As we already noted, there exist real profiles that are very specialized and are not the result of dissociation.

At the same time, we assume that the auxiliary knowledge available to the adversary is limited. It is well known that to achieve perfect privacy against an adversary with unlimited background knowledge is a hard problem [40]. We must clarify that our model does not protect against such an adversary and assumes that the auxiliary information is bounded.

We also assume that the search engine and the agent use the same taxonomy for personalization. This assumption does not hold in most of the cases because search engines' taxonomies are oriented to advertising.[2] It is likely that if we dissociate independently of the search engine's taxonomy, the utility provided by the system will drop. However, since we do not evaluate the utility preserved by the agent, we cannot confirm such effect and leave this evaluation for future work.

Another limitation is that the agent might be considered to break the terms and conditions of the search service. We have implemented DisPA using Google's search engine and we have not found any conflict with their privacy policy. However, since the agent is scraping the URLs of the result list to avoid redirection through Google's servers, it might be argued that the agent is altering the service provided by Google. Furthermore, we do not display Google's advertisements in DisPA results pages.

We note that there is a trade-off between the usability and the privacy offered by the PET for private search. For instance, a search through an anonymous communication system such as Tor would provide stronger privacy guarantees than DisPA. Nevertheless, in terms of overhead, DisPA takes 2.5 s to return results in the worse case, when there is no context created (4 times more than a direct search) and 1.5 s for the average case, in case the context already exists (2 times overhead). The agent also caches result pages to speed up queries that are submitted multiple times over time and, also, prevent the adversary to extract information from the frequency of these queries. We think it is reasonable for a user to sacrifice a second of his search time for a better privacy.

[1] Results of this study can be found at http://www.iiia.csic.es/~mjuarez/results.html.

[2] https://www.google.com/settings/ads

In order to enforce a reproducible research policy, we have uploaded our code to a public repository.[3]

7 Conclusions and Future Work

The main contribution of this research is a framework for the development and evaluation of an agent that provides less disclosure risk in search engines with an admissible time of response. However, DisPA has some drawbacks that future research may deal with. Future work on this line could focus on improving personalized classification. For instance, clicking information could also be incorporated to our model of PQC.

We could also consider vertical searching for personalization. Vertical searching refers to the process of refining consecutive queries to improve search results. The aforementioned PQC should then take into account short-term preferences within a session. One of the issues that arises is how to implement a model for sessions of related queries.

In the line of decreasing disclosure risk, one could consider the generalization of Named Entities using an ontology like WordNet. For instance, if someone searches for "lilburn dentists" the agent could generalize "Lilburn" to "Atlanta" or "Georgia". Information loss would be greater but then it would be possible to measure it by the differences between search results pages.

Along with that, evaluating personalization is still an open problem. There are few studies that aim to measure to what extent search engines personalize search results. However, personalization is a moving target and the authors of these studies often admit that their results are not concluding for not running the experiments for sufficiently long periods of time [41].

Another improvement could be to generalize tracking data used to create virtual identities, from cookies to a more general type of data. Device fingerprinting is still an open problem but there are some promising approaches that could be adopted by DisPA in the future [42].

Besides, the attacking algorithm described in Sect. 5 could be tested with different clustering approaches like sequential clustering described in [43]. The similarity measure could be improved by boosting the tf-idf scheme using a dictionary of terms that differentiates the user from the others.

In addition, other measures of disclosure risk may be defined comparing clusters between clustering of DisPA and non-filtering DisPA. For instance, if dissociated logs in the former fall into several clusters in the latter, disclosure risk is lower than if all fall in the same cluster. The Jaccard index could be used to measure differences between clusters from different classifications.

Finally, we could use more sophisticated privacy measures in our security analysis of the system. For example, we could explore the use of entropy-based measures for this purpose [44, 45].

[3] The source code can be found at https://code.google.com/p/dispa-framework/.

Acknowledgments The authors wish to acknowledge the anonymous reviewers for their detailed and helpful comments to the manuscript. The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n 262608. Partial support by the Spanish MEC projects ARES (CONSOLIDER INGENIO 2010 CSD2007-00004) and COPRIVACY (TIN2011-27076-C03-03) is acknowledged.

References

1. Hansell, S.: Increasingly, internet's data trail leads to court. New York Times Feb (2006)
2. Teevan, J., Dumais, S.T., Horvitz, E.: Personalizing search via automated analysis of interests and activities. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 449–456, ACM (2005)
3. Micarelli, A., Gasparetti, F., Sciarrone, F., Gauch, S.: Personalized search on the world wide web. In: The Adaptive Web, pp. 195–230. Springer, Heidelberg (2007)
4. Norvig, P.: Search Algorithms with Google Director of Research Peter Norvig. Stone Temple Consulting, Oct (2011)
5. Speretta, M., Gauch, S.: Personalized search based on user search histories. In: Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence, pp. 622–628 (2005)
6. Jones, R., Kumar, R., Pang, B., Tomkins, A.: I know what you did last summer: query logs and user privacy. In: Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, pp. 909–914. ACM (2007)
7. EFF: AOL's Massive Data Leak (2009)
8. Sadetsky, G.: AOL Data, Aug (2006)
9. Barbaro, M., Zeller, T.: A Face Is Exposed for AOL Searcher No. 4417749 (2006)
10. Chor, B., Kushilevitz, E., Goldreich, O., Sudan, M.: Private information retrieval. J. ACM (JACM) **45**(6), 965–981 (1998)
11. Peddinti, S.T., Saxena, N.: On the privacy of web search based on query obfuscation: a case study of trackmenot. In: Privacy Enhancing Technologies, pp. 19–37. Springer, Berlin (2010)
12. Juárez, M., Torra, V.: Toward a privacy agent for information retrieval. Int. J. Intell. Syst. **28**, 606–622 (2013)
13. Juárez, M., Torra, V.: A self-adaptive classification for the dissociating privacy agent. In: PST2013, the Eleventh Annual Conference on Privacy, Security and Trust, (Tarragona), pp. 44–50 (2013)
14. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: single database, computationally-private information retrieval. In: Proceedings of the 38th Annual Symposium on Foundations of Computer Science (1997)
15. Yu, S., Thapngam, T., Wei, S., Zhou, W.: Efficient web browsing with perfect anonymity using page prefetching. In: Hsu, C.-H., Yang, L., Park, J., Yeo, S.-S. (eds.) Algorithms and Architectures for Parallel Processing, Lecture Notes in Computer Science, vol. 6081, pp. 1–12. Springer, Heidelberg (2010)
16. Ogata, W., Kurosawa, K.: Oblivious keyword search. J Complexity **20**(2), 356–371 (2004)
17. Ostrovsky, R., Skeith III, W.E.: Private searching on streaming data. In: Advances in Cryptology-CRYPTO 2005, pp. 223–240. Springer, Berlin (2005)
18. Domingo-Ferrer, J., Solanas, A., Castellà-Roca, J.: h(k)-Private Information Retrieval from Privacy-Uncooperative Queryable Databases (2008)
19. Domingo-Ferrer, J., Bras-Amorós, M., Wu, Q., Manjón, J.: User-private information retrieval based on a peer-to-peer community. Data. Knowl. Eng. **68**, 1237–1252 (2009)
20. Shapira, B., Elovici, Y., Meshiach, A., Kuflik, T.: PRAW: A PRivAcy model for the Web. J. Am. Soc. Inf. Sci. Technol. **56**, 159–172 (2005)
21. Murugesan, M., Clifton, C.: Plausibly deniable search. In: Workshop on Secure Knowledge Management vol. 1, pp. 3–8 (2008)

22. Ye, S., Wu, F., Pandey, R., Chen, H.: (2009) Noise injection for search privacy protection. In: 2009 International Conference on Computational Science and Engineering, pp. 1–8 (2009)
23. Howe, D., Nissenbaum, H.: TrackMeNot: Resisting surveillance in web search. In: Lessons from the Identity Trail: Anonymity. Oxford University Press, Oxford (2009)
24. Rebollo-Monedero, D., Forne, J.: Optimized Query Forgery for Private Information Retrieval. IEEE Trans. Inf. Theory **56**, 4631–4642 (2010)
25. Pang, H., Xiao, X., Shen, J.: Obfuscating the topical intention in enterprise text search. In: 2012 IEEE 28th International Conference on Data Engineering (ICDE), pp. 1168–1179. IEEE (2012)
26. Jiménez, J.E., Hoyos, A.R., Parra-Arnau, J., Forné, J., Rebollo-Monedero, D.: Medición de la Privacidad de Perfiles de Usuario mediante un Add-on de Navegador, pp. 93–100 (2013)
27. Balsa, E., Troncoso, C., Diaz, C.: OB-PWS: Obfuscation-Based Private Web Search. In: 2012 IEEE Symposium on Security and Privacy (SP) (2012)
28. Dingledine, R., Mathewson, N., Syverson, P.: Tor: The second-generation onion router. Technical report, DTIC Document (2004)
29. Viejo, A., Sanchez, D.: Providing useful and private web search by means of social network profiling. In: 2013 Eleventh Annual International Conference on Privacy, Security and Trust (PST), pp. 358–361, July 2013
30. Sánchez, D., Castellà-Roca, J., Viejo, A.: Knowledge-based scheme to create privacy-preserving but semantically-related queries for web search engines. Inf. Sci. **218**, 17–30 (2013)
31. Arampatzis, A., Efraimidis, P.S., Drosatos, G.: A query scrambler for search privacy on the internet. Inf. Retr. **16**(6), 657–679 (2013)
32. Erola, A., Castellà-Roca, J.: Using search results to microaggregate query logs semantically. In: DPM/SETOP, pp. 148–161 (2013)
33. Batet, M., Erola, A., Sánchez, D., Castellà-Roca, J.: Utility preserving query log anonymization via semantic microaggregation. Inf. Sci. **242**, 49–63 (2013)
34. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: IEEE Symposium on Security and Privacy, SP 2008, pp. 111–125. IEEE (2008)
35. Google: Key Terms—Policies and Principles, Apr 2012
36. Toubiana, V., Nissenbaum, H.: Analysis of google logs retention policies. J. Priv. Confid. **3**(1), 3–6 (2011)
37. Acar, G., Juárez, M., Nikiforakis, N., Diaz, C., Gürses, S.F., Piessens, F., Preneel, B.: FPDetective: Dusting the web for fingerprinters. In: Proceedings of the 20th ACM Conference on Computer and Communications Security (CCS'13), Berlin, pp. 1129–1140. ACM (2013)
38. Ullegaddi, P., Varma, V.: A Simple Unsupervised Query Categorizer for Web Search Engines (2011)
39. Cao, B., Sun, J., Xiang, E., Hu, D.: PQC: personalized query classification. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 1217–1225 (2009)
40. Ohm, P.: Broken promises of privacy: Responding to the surprising failure of anonymization. UCLA Law Review (2010)
41. Hannak, A., Sapiezynski, P., Molavi Kakhki, A., Krishnamurthy, B., Lazer, D., Mislove, A., Wilson, C.: Measuring personalization of web search. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 527–538. International World Wide Web Conferences Steering Committee (2013)
42. Nikiforakis, N., Joosen, W., Livshits, B.: Privaricator: Deceiving fingerprinters with little white lies. Technical report (2014)
43. Miyamoto, S., Arai, K.: Different sequential clustering algorithms and sequential regression models. In: 2009 IEEE International Conference on Fuzzy Systems, Aug 2009, pp. 1107–1112. IEEE (2009)
44. Serjantov, A., Danezis, G.: Towards an information theoretic metric for anonymity. In: Privacy Enhancing Technologies, pp. 41–53. Springer, Heidelberg (2003)
45. Diaz, C., Seys, S., Claessens, J., Preneel, B.: Towards measuring anonymity. In: Privacy Enhancing Technologies, pp. 54–68. Springer, Heidelberg (2003)

A Survey on the Use of Combinatorial Configurations for Anonymous Database Search

Klara Stokes and Maria Bras-Amorós

Abstract The peer-to-peer user-private information retrieval (P2P UPIR) protocol is an anonymous database search protocol in which the users collaborate in order to protect their privacy. This collaboration can be modelled by a combinatorial configuration. This chapter surveys currently available results on how to choose combinatorial configurations for P2P UPIR.

1 Anonymous Database Search

Privacy issues appear when users query a database. If the concern is regarding the privacy of the content of the query, then the cryptographic solution is called private information retrieval (PIR). A PIR protocol allows the user to retrieve an item from a server without revealing (to the server) which item is retrieved (query privacy). If instead the concern is regarding the privacy of the identity of the querying entity, then the problem is of another nature, and can be solved by a protocol for what is called anonymous database search (sometimes User-Private Information Retrieval-UPIR). Such a protocol allows the user to retrieve an item from a server without revealing (to the server) who is retrieving the item (query anonymity). Observe that PIR and anonymous database search have completely distinct objectives. Typically, a PIR protocol will not achieve query anonymity, nor will a protocol for anonymous database search give query privacy.

Note that available PIR protocols have high computational cost [1]. Therefore it is interesting to evaluate carefully the privacy requirements in the specific case, to see if these can be met by a anonymous database search protocol with smaller

K. Stokes (✉)
Universitat Oberta de Catalunya, Barcelona, Spain
e-mail: kstokes@uoc.edu

M. Bras-Amorós
Universitat Rovira i Virgili, Tarragona, Spain

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_22

computational cost. Also, most PIR protocols only allow the user to query items for which she already knows the position in the database. This fact, together with the high computational cost, makes PIR unsuitable when querying a web-based search engine.

Mixing is another system that provides anonymous networking by anonymizing the origin or the trajectory of the query, or, more in general, the trajectory of any collection of bits travelling over the Internet. However, the use of cookies perishes this anonymity; a cookie is installed by the server on the users computer and allows the server to keep track on all movements the user makes. Also the browser configuration can be used to univocally identify the users.

Other relevant examples of systems for anonymous communications (some are implemented software) are onion routing, Crowds, Tarzan, TrackMeNot, the useless user profile (UUP) protocol, Goopir and Privacy preserving keyword search.

2 A Protocol for Anonymous Database Search

In [2, 3] a protocol that provides anonymous database search was described. The protocol was called peer-to-peer user-private information retrieval (P2P UPIR). The users of the P2P UPIR protocol hide their query profiles by posting each other's queries. For this purpose they form a peer-to-peer (P2P) network over which they share the queries they want to post and the answers to the queries that they have posted. The network uses communication spaces, a memory space together with a cryptographic symmetric key, to share the queries and the answers. The user uploads his encrypted query to a communication space, then a user (another or the same) downloads and decrypts the query, posts it to the web-based search engine, awaits the answer and finally posts the encrypted answer to the same communication space so that the original user can decrypt it and read it. We say that the set of queries that the user u posts to the communication spaces is *the real profile* $RP(u)$ of the user and that the set of queries that the user posts to the server is *the apparent profile* $AP(u)$ of the user. We can model this situation by using an incidence structure.

An incidence structure is a set of points and a set of blocks together with an incidence relation. When every pair of points is contained in at most one block, then the blocks are called lines (the line spanned by two points is the intersection of all blocks through these points).

A combinatorial configuration is an incidence structure such that every point is on r lines, every line contains k points and through every two points there is at most one line or, equivalently, every two lines intersect in at most one point. Combinatorial configurations are also called partial linear spaces. For general references on combinatorial configurations, see [4–6].

We map the users of the P2P UPIR protocol to the points and the communication spaces to the lines of the combinatorial configuration. The users then have access (the key) to the communication spaces that correspond to the lines that pass through the point that represents the user. The reason why we use combinatorial configurations,

and not other incidence structures, is that we want two users to share only one communication space. If they shared two communication spaces, then the information that they share would be doubled.

Within this geometric model, we say that users that share communication spaces are collinear, just as points that are on the same line are collinear. The set of users that are collinear with the user u is called *the neighborhood* $N(u)$ of the user u.

Advantages with the P2P UPIR protocol compared to other systems and protocols are for example that the users can implement the protocol without any collaboration from the database server, that it is suitable for keyword searches and that the complexity is reasonable [2, 3]. Also the P2P UPIR is suitable for complex searches.

This chapter surveys currently available results on how to choose combinatorial configurations for the P2P UPIR protocol. It is a modified and extended version of the article [7], which is otherwise unavailable.

3 Maximal Diffusion of the Real Query Profile

It is clear that the effect of the protocol described in Sect. 2 is to diffuse the real query profiles of the users into the apparent profiles of the users in the neighborhoods of the users. Therefore, it is interesting to maximize the size of the neighborhoods of the users. The point set of the combinatorial configuration that we use should have the same cardinality as the number of users of the protocol. Hence, once the number of points is fixed, we want a combinatorial configuration that maximizes the neighborhoods of the points. A configuration with only two points on each line is a graph. It is clear that the family of graphs that maximize the size of the neighborhoods of the points are the complete graphs. In a complete graph, any pair of vertices is connected by an edge. A linear space is a configuration in which any pair of points is connected by a line. In a linear space the neighborhood of the point p is the whole point set except for p. Therefore, in a linear space the neighborhoods of the points are as large as possible in a combinatorial configuration with v points, so in this sense the linear spaces are the optimal choice of combinatorial configuration for P2P UPIR.

Theorem 1 *In terms of largest neighborhoods, the optimal configurations for P2P UPIR are the linear spaces.*

Examples of linear spaces with $k > 2$ are the finite projective planes over the finite field $\mathbb{F}_q$ with $k = q + 1$ and $v = q^2 + q + 1$, and the finite affine planes over $\mathbb{F}_q$ with $k = q$ and $v = q^2$, see Fig. 1.

4 Avoiding Collusions of Adversaries

Consider a community of users that are implementing an instance of a P2P UPIR protocol that takes as parameter a combinatorial (r, k)-configuration C. The community

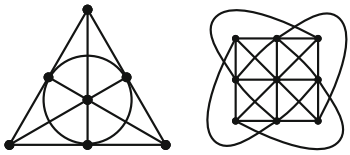


Fig. 1 The projective plane over $\mathbb{F}_2$ and the affine plane over $\mathbb{F}_3$

of users is mapped to the points in C and the users are assigned communication spaces that correspond to the lines of C. A user u_0 shares his queries with the users in the neighborhood $N(u_0)$. For u_0, to share his queries implies a privacy risk.

4.1 Collusions of Peers Allowing Communications Over Any Channel

In this section we treat the situation where there is a set of colluding users who can communicate over channels that are not controlled by the protocol. This can for example be assumed to be the case when an adversary controls a set of users which he introduced in the user set for this purpose. We let the knowledge of such an adversary model the knowledge of a collusion of adversaries in general.

Suppose that $u_1 \in N(u_0)$ is a user of the protocol who shares a communication space c with u_0. Some versions of the P2P UPIR protocol require u_1 to read c only when he has a query of his own, other versions let u_1 read c on synchronized intervals. In any case, it is clear that if u_1 wants to read all the queries that u_0 puts on c, this is possible. We may therefore assume that u_1 has access to all queries that u_0 uploads to c. These queries form a proportion of $1/r$ of the whole set of u_0's queries, that is, of u_0's real profile $RP(u_0)$. However, on the communication space c there are queries from k different users and the queries from u_0 are mixed with the other queries. Therefore u_1 does not know to whom of the $k-1$ users different from himself the queries on c belong.

An adversary who owns all users on c except for u_0 will however know which of the queries belong to u_0. This observation suggests a strategy for an adversary who wants to access the real profile of u_0, consisting in introducing users in the protocol such that they are collinear with u_0 and such that they are all on the same line. In order to completely control one of u_0's communication spaces, the adversary has to introduce $k-1$ users on one line which goes through u_0. In order to completely control m of u_0's communication spaces, the adversary must introduce $m(k-1)$ users on m lines which all go through u_0.

Lemma 2 *An adversary who controls $k-1$ users on the same line through u_0 has complete control over a proportion of $1/r$ of the real profile of u_0. If he controls $m(k-1)$ users on m lines through u_0, then he has complete control over a proportion of m/r of the real profile of u_0.*

Indeed, if the adversary controls all other users who signed up to implement the protocol, then the adversary will know the entire real profile of u_0. We will assume that it is difficult for the adversary to introduce large quantities of colluding users. If the adversary wants to have access to the largest possible proportion of the queries in the real profile of u_0, but does not care if this profile is mixed with queries from other users (the adversary might employ traffic analysis to know the origin of the query), then he will be more interested in introducing the colluding users such that they are collinear with u_0 by different lines. In this way the adversary will only need m users in order to have access to a proportion of m/r of the queries in the real profile of u_0, although these queries will be mixed with the queries of other users.

Lemma 3 *An adversary who employs traffic analysis and controls m users on different line through u_0 has control over a proportion of m/r of the real profile of u_0.*

4.2 Avoiding Collusions of Curious Peers Communicating Over the Protocol Channel: Triangle-Free Configurations

Assume now that any set of colluding users can communicate only over the channels provided by the protocol, that is, over the communication spaces to which they have access.

Let U be a set of users implementing an instance of a P2P UPIR protocol with a combinatorial configuration C. Consider the users u_0, u_1 and u_2 in U. Suppose that u_1 and u_2 want to form a collusion with the aim to obtain an advantage over the protocol and get access to a larger proportion of the real profile of u_0 than the protocol normally permits. If u_1 and u_2 share two different communication spaces with u_0, then the quantity of queries from the real profile of u_0 accessible to u_1 and u_2 together, is twice the quantity accessible to u_1 and u_2 on their own. In the geometric language we used before, we say that u_1 and u_2 are collinear to u_0 by two different lines, say l_1 and l_2, the lines that correspond to the two different communication spaces they share with u_0.

Since we have assumed that all the communication between the users must be done over the communication channels provided by the protocol, in order for u_1 and u_2 to share their information on u_0 they must have access to a common communication space. That is, u_1 and u_2 must be collinear, say by the line l_3. We see that u_0 can not be on the line l_3. Indeed if u_0 was on l_3, then the pair of points u_0 and u_1 would be both on l_1 and l_3, so that $l_1 = l_3$. Also the pair of points u_0 and u_2 would be both on l_2 and l_3, so that $l_2 = l_3$. But we have supposed $l_1 \neq l_2$, so this is absurd. We deduce that l_1, l_2 and l_3 form a triangle in C.

Lemma 4 *The use of a triangle-free configuration for the P2P UPIR protocol avoids collusions of two users communicating over the channels provided by the protocol.*

The previous arguments can be generalized to a set of n colluding users. Suppose that a set of n users want to form a collusion to obtain as much as possible of the real profile of u_0 and that they only have access to the communication channels provided by the P2P UPIR, that is, to the communication spaces. From the previous discussion it is clear that the n users should all be collinear to u_0. We also previously saw that the users can be either

1. collinear with u_0 on the same line,
2. collinear with u_0 by different lines and finally, for $n > 2$ users,
3. both of the previous situations can occur.

Suppose that the adversary introduces n colluding users in the protocol. Then he obtains access to the largest proportion of the real profile if the colluding users are introduced so that they are collinear with u_0 by different lines. On the other hand if the colluding users are introduced on the same line, then the adversary obtains better control of which queries on the communication space that pertain to the real profile of u_0. Suppose that the former type of control is more interesting to the adversary than the latter. That is, suppose that the adversary wants to introduce the colluding users so that they are collinear with u_0 by different lines.

In order for these users to communicate they need to share communication spaces, that is, they need to be collinear. The best communication is obtained if they are pairwise collinear, that is, if every pair of users in the set of colluding users shares a communication space. Following the same arguments as in the case of two colluding users, it is easy to see that this requires the existence of a triangle through every triple of points u_0, u_i, u_j where u_i and u_j are colluding users. A simple counting argument then shows that the number of required triangles through u_0 is $n^2/2$. The highest proportion of the real profile of u_0 which can be read in this way requires a set of r colluding users, one sitting on every line through u_0. The number of triangles through u_0 required in this case is $r^2/2$. In this constellation the r colluding users can indeed read the entire real profile of u_0, although it will be mixed with queries from other users. One type of combinatorial configuration which permits this attack are the finite projective planes, in which every three points are on a triangle.

One can imagine a more sparse constellation of colluding users that may require less triangles. For example, n colluding users $\{u_i\}_{i=1}^{n}$ may be located so that they are all collinear with u_0 and connected in between the collusion only by, say, one path of lines $\{l_i\}_{i=1}^{n-1}$, so that the line l_i is spanned by the points assigned to the users u_i and u_{i+1}. In any case, all these constellations of colluding users are avoided if the combinatorial configuration used in the P2P UPIR protocol is triangle-free. See for example [8] for the existence and construction of triangle-free combinatorial configurations.

Remember though that if communication between colluding users is permitted also on a channel external to the protocol then it does not matter whether they are on a triangle or not. In general, for n colluding users to obtain access to n times the information on u_0 as the protocol would permit, they must be collinear with u_0 by n different lines. Calculations of the probability of this event to happen can be found in [9].

5 Reidentifying Users Through Their Neighborhoods

In this section we describe how configurations with unique neighborhoods may be problematic in some implementations of P2P UPIR. We also describe how to avoid this problem through the use of configurations with anonymous neighborhoods.

5.1 Combinatorial Configurations for P2P UPIR v1

In the first version of P2P UPIR the users did not post any of their own queries to the server (we will call this protocol P2P UPIR v1). This behaviour is problematic; we will now see that by choosing this strategy the user already reveals some information to the server.

Suppose that a set of users U are implementing the P2P UPIR protocol and fix a user u_0 in U. Then the users in U that post the real profile $RP(u_0)$ to the server are the users in $N(u_0)$, that is, $RP(u_0) \subset \bigcup_{v \in N(u_0)} AP(v)$. Suppose that the user u_0 posts the same query several times and that this query is not too common. Then this query will be in the apparent profiles $AP(v)$ for users $v \in N(u_0)$, but not in the apparent profiles of other users in U. Hence, if we can map the neighborhood $N(u_0)$ to u_0, then we can also map the query to the user u_0.

One can ask if the repetition of queries is a common phenomenon among the users of web-based search engines. Indeed, [10] discusses common user situations that provoke repetitions of queries and the results presented in [11, 12] show that repetitions of queries is a frequent and common behaviour.

The risk is the combination of repetition of queries together with a mapping $N(u_0) \mapsto u_0$. Since it is hard to change the users tendency to repeat queries, we concentrate on solving the other aspect of the problem. We will divide this task into two parts: identifying the problematic combinatorial configurations and finding suitable combinatorial configurations that avoid the problem.

5.1.1 Combinatorial Configurations with a Bijection Between the Point Set and the Neighborhood Set

The problematic combinatorial configurations are the ones for which the map $p \mapsto N(p)$ is invertible. We ask the question: exactly which are these combinatorial configurations? A partial solution to this problem is given by the following theorem.

Theorem 5 *Consider a combinatorial (r, k)-configuration with point set P that is either a linear space or a triangle-free configuration with $k > 2$. Then the mapping*

$$\begin{aligned} P &\to \{N(p) : p \in P\} \\ p &\mapsto \quad N(p) \end{aligned}$$

is a bijection.

Therefore these two types of combinatorial configurations are problematic when used with the P2P UPIR v1 protocol. This is important, because we have seen that the triangle-free configurations are suitable in order to avoid collusions of curious peers and the linear spaces are optimal in the sense that they are the combinatorial configurations that diffuse the real profile of a user u into the apparent profiles of all other users of the protocol but the user u.

5.1.2 Combinatorial Configurations with n-Anonymous Neighborhoods

We have seen examples of combinatorial configurations that imply a privacy risk when used for the P2P UPIR v1 protocol if there are repetition of queries. The reason why they imply a privacy risk is that all the points have a unique neighborhood. Now we will look at combinatorial configurations that avoid this problem, combinatorial configuration with n-anonymous neighborhoods.

The concept of n-anonymity appeared in the study of disclosure risk control for statistical databases. A database (table) is a collection of records that correspond to individuals and that can be divided into attributes. It is normally assumed that some of these attributes belong to the public knowledge, while other attributes contain sensitive information that should be protected so that it can not be linked to the individual behind the record. An identifier in a table is an attribute that uniquely identifies the individuals.

Removing the identifiers is usually not enough to protect the sensitive information in the table. Records can be linked to individuals by their entries in a collection of attributes, which together identify the individual. A collection of attributes that permits the identification of at least one individual is called a *quasi-identifier*. A quasi-identifier is normally determined apriori, using general information on the table structure and the investigated population. A table in which every collection of entries in a quasi-identifier is repeated at least n times is an n-anonymous table. In the literature, the concept of n-anonymity is usually called k-anonymity, see [13–15].

In our case the neighborhoods of the points can be regarded as quasi-identifiers in the database that contains the apparent profiles of the users of the P2P UPIR protocol. The solution would be to find combinatorial configurations that are n-anonymous with respect to this quasi-identifier.

Definition 6 A combinatorial configuration provides n-anonymous P2P UPIR v1 when every point shares its neighborhood with at least $n - 1$ other points.

It is clear that such a combinatorial configuration has a partition of the point set such that all points in the same part share neighborhood. A nice example of combinatorial configurations that provide n-anonymous P2P UPIR v1 are the tranversal designs.

Definition 7 A transversal design $TD_\lambda(k, n)$ is an incidence structure with point set P and a block set such that

- $|P| = nk$,
- every block contains k points,
- there is a partition of the point set in k parts (called groups) of size n,
- any group and any block contain exactly one common point and
- every pair of points from distinct groups is contained in exactly λ blocks.

A transversal design is a combinatorial configuration if and only if $\lambda = 1$.

Theorem 8 *A transversal design $TD_1(k, n)$ provides n-anonymous P2P UPIR.*

The transversal designs are not the only combinatorial configurations that provide n-anonymous P2P UPIR v1, but they are regular and easy to construct and therefore suitable for applications. Indeed, if we want regularity and if we maximize k, then the n-anonymous combinatorial configurations are exactly the transversal designs [16].

5.2 Combinatorial Configurations for P2P UPIR v2

We have seen that it is possible to find combinatorial configurations that avoid the risk caused by the repetition of queries. The risk was caused by the fact that the real query profile was not homogeneously diffused into the apparent profiles of the set of users that consist of the user u together with his neighborhood $N(u)$. This was due to the choice of the user not to post any of his own queries to the server. It can be proved that if the user chooses to post also his own queries when he finds them on the communication spaces, the problem persists, although in this case the user will end up posting a larger proportion of his own queries than will the other users.

We can modify the protocol in order to spread the user's real query profile homogeneously over $N(u) \cup \{u\}$. The solution is given by the following theorem.

Theorem 9 *Consider a community of users implementing a P2P UPIR protocol with a combinatorial (v, b, r, k)-configuration and impose on the users to check their communication spaces with a fixed frequency that is higher or equal to the frequency with which they post queries (checking the communication spaces is equivalent to posting a garbage query.) Then the user u's real profile is optimally diffused into the apparent profiles of $N(u) \cup \{u\}$ if u forwards a proportion of*

$$\frac{1}{r(k-1)+1}$$

of his own queries to the server.

Observe that the quantity $r(k-1)+1$ equals $\sharp(N(u) \cup \{u\})$. After modifying the P2P UPIR v1 protocol according to the recommendations stated in Theorem 9, the new version of the protocol is called P2P UPIR v2 [10]. Now the set of users that "emits" the real profile of the user u is the closed neighborhood $N(u) \cup \{u\}$. Therefore,

to attain n-anonymity in P2P UPIR v2 it is interesting to use a configuration for which the map $p \mapsto N(p) \cup \{p\}$ has $n > 1$ preimages for each point p.

Definition 10 We say that a combinatorial configuration with point set P provides n-anonymous P2P UPIR v2 if for every point $p \in P$ there are at least n distinct points $p_i \in P, i \in \{1, \ldots, n\}$ with

$$N(p_i) \cup \{p_i\} = N(p) \cup \{p\}.$$

Configurations for which the map $p \mapsto N(p) \cup \{p\}$ is injective, so that they have unique closed neighborhoods, are bad configurations for the P2P UPIR v2 protocol. Examples of such configurations are configurations with deficiency one, so that each point is collinear with all points but one. Also the transversal designs have this property. For some further examples see [16].

The modification of the protocol was made with the linear spaces in mind. As we saw in Sect. 3, the linear spaces have the largest neighborhoods among all configurations with the same number of points, and so they are optimal in terms of maximal diffusion of real query profile. This makes them attractive, once the vulnerability for repeated queries is removed. The next result shows that the modification of the protocol achieved exactly this.

Theorem 11 *A linear space on v points provides n-anonymous P2P UPIR v2 with $n := v$. Since v is the number of users implementing the protocol, this is optimal.*

Because of the similarity between the definition of n-anonymous P2P UPIR v1 and n-anonymous P2P UPIR v2, we can construct combinatorial configurations for n-anonymous P2P UPIR v2 from the combinatorial configurations for n-anonymous P2P UPIR v1.

Theorem 12 *Let C be a combinatorial (v, b, r, k)-configuration with $k|n$ that provides n-anonymous P2P UPIR v1, so that every point shares neighborhoods with* exactly *n more points. Then there also exists a combinatorial $(v, b + n, r + 1, k)$-configuration C' that provides k-anonymous P2P UPIR v2.*

In these combinatorial configurations the sets $N(u) \cup \{u\}$ are in general smaller than in a linear space (assuming they have the same number of poins), so they are suboptimal with respect to the diffusion of the real profiles of the users. Affine planes can be constructed in this way from certain transversal designs, and affine planes are linear spaces.

6 Using Other Designs

In the previous sections we have seen examples of configurations which belong to larger families of designs. The transversal designs were already introduced with the terminology of design theory, but also the linear spaces belong to a well-known

family of designs. A $t-(v,k,\lambda)$ design is a set of points and a set of subsets of the point set called blocks, such that any t points appear together in exactly λ blocks. The linear spaces are exactly the $2-(v,k,1)$ designs.

More generally, a design, or a set system, is a finite set of points X together with a family B of—not necessarily distinct—subsets of X called blocks. So, the points and the lines of any configuration form a design.

The use of designs with $\lambda \neq 1$ for P2P UPIR was introduced by Swanson and Stinson in [17], preceeded by a modification of the protocol which separates the key distribution from the proxy assignment. In their protocol, the user first assigns a proxy for the query and subsequently uploads the query to a communication space that is shared by the user and the proxy, encrypting the query with the corresponding cryptographic key. The proxy is selected with uniform distribution from the total set of users, in order to achieve perfect anonymity against the database server. The effect on the real profile of the user is similar to what is achieved by P2P UPIR v2 with a configuration with v points and v-anonymous closed neighborhoods. We have seen that this is optimal in terms of diffusion of the query profile, so this is an attractive solution.

In Swanson and Stinson's protocol, query anonymity against the server is not affected if two users share more than one communication space. Therefore the use of designs in which two points are in more than one block does not complicate the analysis. However, a requirement for the protocol to work is that any two users can communicate over at least one communication space. Note that the only combinatorial configurations with this property are the linear spaces. As Swanson and Stinson point out, in their protocol, for achieving query anonymity against the server it is enough to use a covering design, a design in which every two points are in at least one block. In a covering design, the blocks can contain different numbers of points, and the points can appear in different numbers of blocks. Therefore the use of covering designs allows a great flexibility, compared to the use of for example a t-design. However, as also noticed by Swanson and Stinson, the query anonymity against other users then depends on the combinatorial properties of the covering design. They propose the use of regular pairwise balanced designs (PBD) to meet the problem of maintaining privacy also against other users. A PBD is a design in which every pair of distinct points is contained in exactly λ blocks. It is regular if all points are contained in the same number of blocks. A PBD is clearly a covering design. More in general, for query anonymity against collusions of users, they propose the use of covering designs with permanent t-anonymity sets for P2P UPIR.

7 Conclusions and Future Work

We have surveyed the state of the art of the use of combinatorial configurations for P2P UPIR. The protocol is primarily designed to give query anonymity against the database server. We have seen that this purpose is achieved by the P2P UPIR v1 protocol when using a configuration with n-anonymous open neighborhoods and by

the P2P UPIR v2 protocol with a configuration with n-anonymous closed neighborhoods. Examples of the former are the transversal designs and examples of the latter are the linear spaces. If the user selects the proxy for each query uniformly from the set of users, as done by Swanson and Stinson, the linear spaces may be replaced by the t-designs. In general, this approach allows the use of covering designs for P2P UPIR.

In practice, assigning a specific proxy to each query can be problematic if the selected proxy is currently unavailable. In this sense a protocol that instead assigns to each query a set of proxies on a communication space, like the original protocol, can be more robust. However, it should then be remembered that the unavailability of proxies may weaken the provided anonymity.

Once query anonymity against the database server is achieved, it is important to assess query anonymity also against other users. Some solutions in this line have already been proposed, as surveyed here, but in general this is still a promising subject for future research.

Multi-hop P2P UPIR has been proposed for the sake of providing anonymity against other users, see for example [17, 18]. It is our belief that the analysis of such a system should be similar to the analysis of the Crowds system [19]. It is interesting to note that the use of configurations and designs in the P2P UPIR protocol causes it to use less cryptographic keys than the Crowds system does. Indeed, one of the problems with the Crowds system is the large amount of required keys, caused by the use of a key-distribution that is defined by a complete graph.

Acknowledgments The authors acknowledge financial support from the Spanish MEC projects ARES (CONSOLIDER INGENIO 2010 CSD2007-00004) and ICWT (TIN2012-32757).

References

1. Ostrovsky, R., Skeith III W.E.: A survey of single-database private information retrieval: techniques and applications. In: Proceedings of the 10th International Conference on Practice and Theory in Public-Key Cryptography, pp.393411. Springer, Berlin (2007)
2. Domingo-Ferrer, J., Bras-Amorós, M.: Peer-to-Peer Private Information Retrieval. Lecture Notes in Computer Science, Privacy in Statistical Databases (2008)
3. Domingo-Ferrer, J., Bras-Amorós, M., Wu, Q., Manjón, J.: User-private information retrieval based on a peer-to-peer community. Data Knowl. Eng. **68**(11), 1237–1252 (2009)
4. Gropp H.: Configurations. In: 2nd Edn. Colbourn, C.J., Dinitz, J.H. (eds.) The CRC Handbook Of Combinatorial Designs, pp. 352–355. CRC Press, Boca Raton, FL, (2007)
5. Grünbaum, B.: Configurations of Points and Lines. American Mathematical Society, Providence, RI (2009)
6. Pisanski, T., Servatius, B.: Configurations from a Graphical Viewpoint. Birkäuser Advanced Texts, Basler Lehrbücher, Springer (2013)
7. Stokes, K., Bras-Amorós, M.: Combinatorial structures for an anonymous data search protocol. In: Proceedings of Workshop on Computational Security, CRM (UAB), Barcelona, November 28 Dec 2 (2011)
8. Stokes, K., Bras-Amorós, M.: Associating a numerical semigroup to the triangle-free configurations. Adv. Math. Commun. **5**(2), 351–371 (2011)

9. Stokes, K.: Combinatorial structures for anonymous database search. Doctoral thesis. Universitat Rovira i Virgili (2011)
10. Stokes, K., Bras-Amorós, M.: On query self-submission in peer-to-peer user-private information retrieval. In: Proceedings of the 4th International Workshop on Privacy and Anonymity in the Information Society (PAIS'11), Uppsala, Sweden (2010)
11. Spink, A., Wolfram, D., Jansen, M.B.J., Saracevic, T.: Searching the web: the public and their queries. J. Am. Soc. Inform. Sci. Technol. **52**(3), 226–234 (2001)
12. Teevan, J., Adar, E., Jones, R., Potts, M.: History repeats itself: repeat queries in Yahoo's query logs. In: Proceedings of the 29th Annual ACM Conference on Research and Development in Information Retrieval (SIGIR'06). pp. 703–704. (2005)
13. Samarati, P.: Protecting respondents' identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)
14. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. SRI International Technical Report (1998)
15. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. of Unc., Fuzz. Knowl. Based Syst. **10**(5), 557–570 (2002)
16. Stokes, K., Farrás, O.: Linear spaces and transversal designs: k-anonymous combinatorial configurations for anonymous database search. Des. Codes Crypt. **71**(3), 503–524 (2014)
17. Swanson, C.M., Stinson, D.R.: Extended combinatorial constructions for peer-to-peer user-private information retrieval. Adv. Math. Commun. **6**, 479–497 (2012)
18. Domingo-Ferrer, J., González-Nicolás, Ú.: Rational behavior in peer-to-peer profile obfuscation for anonymous keyword search: the multi-hop scenario. Inf. Sci. **200**, 123–134 (2012)
19. Reiter, M., Rubin, A.: Crowds: anonymity for web transactions. ACM Trans. Inform. Syst. Sec. **1**(1), 66–92 (1998)
20. Bras-Amorós, M., Domingo-Ferrer, J. Stokes, K.: Configuraciones combinatóricas y recuperación privada de información por pares. In: Congreso de la Real Sociedad Matemática Española-RSME 2009, Oviedo, Spain (2009)
21. Bras-Amorós, M., Stokes, K.: The semigroup of combinatorial configurations. Semigroup Forum, vol. 84, pp. 9196. Springer, New York (2012)
22. Bras-Amorós, M., Stokes, K, Greferath, M.: Problems related to combinatorial configurations with applications to P2P-user private information retrieval. In: Proceedings of The 19th International Symposium on Mathematics with Applications, pp. 15681577. (2010)
23. Bras-Amorós, M., Stokes, K.: On the existence of combinatorial configurations. In: Proceedings of the 3rd International Workshop on Optimal Networks Topologies (IWONT), June 9–11 2010, Barcelona, pp. 145–168. (2010)
24. Stokes, K., Bras-Amorós, M.: Optimal configurations for peer-to-peer user-private information retrieval. Comput. Math. Appl. **59**(4), 1568–1577 (2010)

Part VIII
User Privacy: Recommender and Personalized Systems

Privacy-Enhancing Technologies and Metrics in Personalized Information Systems

Javier Parra-Arnau, David Rebollo-Monedero and Jordi Forné

Abstract In recent times we are witnessing the emergence of a wide variety of information systems that tailor the information-exchange functionality to meet the specific interests of their users. Most of these personalized information systems capitalize on, or lend themselves to, the construction of user profiles, either directly declared by a user, or inferred from past activity. The ability of these systems to profile users is therefore what enables such intelligent functionality, but at the same time, it is the source of serious privacy concerns. The purpose of this paper is twofold. First, we survey the state of the art in privacy-enhancing technologies for applications where personalization comes in. In particular, we examine the assumptions upon which such technologies build, and then classify them into five broad categories, namely, basic anti-tracking technologies, cryptography-based methods from private information retrieval, approaches relying on trusted third parties, collaborative mechanisms and data-perturbative techniques. Secondly, we review several approaches for evaluating the effectiveness of those technologies. Specifically, our study of privacy metrics explores the measurement of the privacy of user profiles in the still emergent field of personalized information systems.

1 Privacy Issues in Personalized Information Systems

Selecting and directing information are crucial in every aspect of our modern lives, including areas as diverse as health, leisure, marketing and research. In the past, these processes were largely manual, but due to the exponential improvements in

J. Parra-Arnau (✉) · D. Rebollo-Monedero · J. Forné
Department of Telematics Engineering, Universitat Politècnica de Catalunya (UPC),
08034 Barcelona, Spain
e-mail: javier.parra@entel.upc.edu; xparnau@gmail.com

D. Rebollo-Monedero
e-mail: david.rebollo@entel.upc.edu

J. Forné
e-mail: jforne@entel.upc.edu

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_23

computation and memory, sophistication of software and the gradual ubiquity of mobile and fixed Internet access, they are now becoming increasingly automated.

The automation of these processes clearly facilitates effective handling of information. In a world where online information systems, society and economics have become inextricably entangled, the automated, personalized filtering and selection of an otherwise overwhelming overabundance of information is indispensable. To put this continuous bombardment of information in numbers, every minute 6,600 pictures are uploaded to Flickr, 600 videos are submitted to YouTube, 70 new Internet domains are registered, 98,000 tweets are generated on the social networking site Twitter, 20,000 new posts are published on the micro- blogging platform Tumblr and 12,000 new ads are posted on Craigslist [1].

Endowing the above systems with intelligent processes for the selection and direction of such tremendous flow of information increases their usability and guarantees their effectiveness. Said processes of information filtering and targeting can be built on the basis of *user profiles*, either explicitly declared by a user, or derived from past activity. Automated information filtering may, for example, help tailor a Google search to the personal preferences of a user, by leveraging on their search history. When searching in Facebook for a name of a person we would like to become virtual friends with, the site takes into account numbers of common friends to recommend the most likely person with that name. Under a conceptual, abstract perspective, personalized search and social networks are really a special case of recommendation systems, which encompass functionality of a growing variety of information services, predominantly multimedia recommendation systems such as YouTube, Netflix, Spotify, the Genius function of iTunes or Pandora Radio, to name just a few.

At the heart of these *personalized information systems* is therefore profiling. From a home computer or a smartphone, users submit queries to Google, search for news on Digg, rate movies at IMDb and tag their favorite Web pages on Delicious. Over time, the collection and processing of all these actions allow such systems to extract an accurate snapshot of their interests or user profile, without which the desired personalized service could not be provided. Profiling is thus what enables those systems to determine what information is relevant to users, but at the same time, it is the source of serious privacy concerns. User profiles may reveal sensitive information such as health- related issues, political preferences, salary and religion, not only about the user in question, but also about other users with whom social relationships are available to the service provider.

The purpose of this paper is to survey the state of the art in privacy-enhancing technologies (PETs) for applications where personalization comes in. In particular, we examine the assumptions upon which such technologies build, and then classify them into five broad categories. Secondly, we review several approaches for evaluating the effectiveness of those technologies. In particular, our study of privacy metrics explores the measurement of the privacy of user profiles in the still emergent field of personalized information systems.

2 Privacy Protection in Personalized Information Systems

In this section, we shall examine the main proposals aimed at protecting user privacy in the scenario of personalized information systems. Before proceeding, Sect. 2.1 will introduce several *trust models*, essentially assumptions about the level of trust that users place in the entities they communicate with. The next subsection, Sect. 2.2, will survey the approaches of the state of the art in this scenario, showing in each case the level of trust assumed by users.

2.1 Trust Models

A number of actors are involved in the provision of personalized services. Among these actors, we obviously find users and the information systems themselves, but also we have the Internet service provider (ISP), routers, switches, firewalls and any other networking infrastructure placed between the service provider and the end user.

Any of these entities may be considered as an attacker. To hinder these attackers in their efforts to compromise user privacy, users have a wide variety of PETs at their disposal, such as the technologies based on proxy systems, protocols exploiting collaboration among users, or mechanisms capitalizing on data perturbation. In some of these cases, users must place all their trust in these technologies. In other cases, however, it is not necessary that users trust the underlying privacy- protecting mechanism. In this section we define three models that specify this degree of trust. Such levels will allow us to identify the assumptions upon which the mechanisms surveyed in Sect. 2.2 build.

In the *trusted model*, users entrust an external entity or trusted third party (TTP) to safeguard their privacy. That is, users put their trust in an entity which will hereafter be in charge of protecting their private data. In the literature, numerous attempts to protect user privacy have followed the traditional method of anonymous communications, which is fundamentally based on the suppositions of our trusted model. Additional examples of PETs assuming this model are anonymizers and pseudonymizers. The idea behind these TTP-based approaches is conceptually simple. Their main drawbacks are that they come at the cost of infrastructure and suppose that users are willing to trust other parties. However, even in those cases where we could trust an entity completely, that entity could eventually be legally enforced to reveal the information they have access to [2]. The AOL search data scandal of 2006 [3] is another example that shows that the trust relationship between users and TTPs may be broken. In short, whether privacy is preserved or not depends on the trustworthiness of the data controller and its capacity to effectively manage the entrusted data.

On the other extreme is the *untrusted model*, where users mistrust any of the aforementioned actors. Since users just trust themselves, it is their own responsibility to protect their privacy. Examples of mechanisms relying on the assumptions of our untrusted model are those based on data perturbation and operating on the user side.

In this kind of data-perturbative approaches, users need not trust any entity but, privacy protection comes at the cost of system functionality and data utility.

On a middle ground lies the *semi-trusted model*, where trust is distributed among a set of peers that collaborate to protect their privacy against a set of untrusted entities. An example of this trust model is found in the collaborative or peer-to-peer (P2P) approaches examined later in Sect. 2.2. In these approaches, users trust other peers and typically participate in the execution of a protocol aimed at guaranteing their privacy. Users clearly benefit from this collaboration, but nothing can prevent a subset of those peers from colluding and compromising the privacy of other users.

2.2 Privacy-Enhancing Technologies

In this section we review the state of the art in PETs in the context of personalized information systems. Partly inspired by [4], we classify these technologies into five categories: basic anti-tracking technologies, cryptography-based methods from private information retrieval (PIR), TTP-based approaches, collaborative mechanisms and data-perturbative techniques. We would like to stress that many of the technologies reviewed, far from being mutually exclusive, may in fact be combined synergically.

2.2.1 Basic Anti-tracking Technologies

A key element in the provision of personalized services are *tracking technologies*. Thanks to these technologies, personalized information systems can identify users across different visits or sessions as well as multiple Web domains. Tracking mechanisms are therefore a means of driving personalization, as they allow these systems to follow users over time, thus enabling profiling.

The inherent operation of the Internet does permit tracking users. As many other data-communication networks, the Internet requires that every user[1] be identified by a unique address, in order for messages to be routed through the network. ISPs are precisely in charge of allocating addresses to users and keeping the correspondence between user identifiers and addresses. In this manner, users wishing to communicate through the Internet just need to attach the source and destination addresses to the message to be sent. On the one hand, these addresses enable the intermediary entities (switches, routers, firewalls) involved in the communication process to forward these messages until the destination address is reached. But on the other hand, since the addresses are transmitted in the clear, the entities themselves or any adversary capable of intercepting the messages may ascertain who is communicating with whom and therefore may track user activity.

[1] Technically, machines, not users, are identified by addresses.

Employing dynamic IP addresses and rejecting hypertext transfer protocol (HTTP) cookies are two basic methods to prevent an attacker, possibly the service provider itself, from tracking users. The identification of users through IP addresses actually fails when a large number of users share a single IP address. This is the case of the users of a private network who resort to network address translation [5] and share a static IP address. The use of the dynamic host configuration protocol [6] also provides a means to hinder privacy attackers in their efforts to monitor user behavior. The main drawback of dynamic IP addresses is that the assignment and renewal of these addresses are controlled by ISPs. On the other hand, rejecting HTTP cookies may be an alternative to avoid tracking. The problem of this approach is that it can disable other Web services.

The result of the application of these basic mechanisms is clear: the attacker cannot build a profile of the user in question, but this is at the expense of a nonpersonalized service; if the service provider is unable to profile users based, for example, on their search or tag history, no personalization is possible. We would like to note that if these methods were completely effective, users would achieve the maximum level of privacy protection, but the worst level in terms of utility. In terms of performance, these mechanisms would be comparable to those more conventional techniques based on access control or encryption. As we shall see in the remainder of this state-of-the-art section, other PETs aimed at preserving user privacy in the context of personalized information systems assume that users are tracked and, in a way, identified. The aim of some these approaches is then to thwart the attacker from *accurately* profile users.

2.2.2 Private Information Retrieval

In this subsection we briefly touch upon a few early proposals in the field of PIR. Afterwards, we review other mechanisms relying also on cryptography. As we shall see, the PETs reviewed in this subsection and the anti-tracking technologies examined above have much in common: both approaches may provide users with the highest level of privacy protection but at the cost of nonpersonalized services.

PIR refers to cryptography-based methods that enable a user to privately retrieve the contents of a database, indexed by a memory address sent by the user, in the sense that it is not feasible for the database provider to ascertain which of the entries was retrieved [7, 8]. In the context of Web search, PIR protocols allow a user to look up information in an online database without letting the database provider know the search query or response. A simple way to provide this functionality is as follows: the database provider submits a copy of the entire database to the user so that they can look up the information themselves. This is known as trivial download. The field of PIR is aimed at transferring less data while still preserving user privacy.

The first PIR protocol [9] traces back to 1995. Said protocol allowed users to privately retrieve records from a series of replicated copies of a database. In this scheme, each of the servers storing a copy of that database could not learn any information about the items retrieved by the user; this was, however, at the expense of a large amount of communication. In the current information systems, the implementation

of this solution is impractical; normally these systems make use of a database stored on a single server. Despite these shortcomings, this initial work triggered numerous and important contributions to the field.

An alternative to this protocol was [10], which proposed the first single-server approach in 1997. As in many subsequent PIR protocols, the main problem with this alternative is that it requires the participation of the server itself. In other words, the single-server approach implicitly assumes that the database provider will have some incentives to help users protect their protect. In practice, this is an unrealistic assumption.

Although the literature of PIR is particularly rich and extensive, the mechanisms proposed so far have several major limitations. First, considering the inherent operation of these protocols, we may conclude that personalization is unfeasible. Since the database provider does not know neither the queries nor the corresponding answers, users cannot be profiled by the provider. And secondly, there are several disadvantages that preclude the practical deployment of these cryptographic methods: PIR protocols require the provider's cooperation, are limited to a certain extent to query-response functions in the form of a finite lookup table of precomputed answers, and are burdened with a significant computational overhead. A comprehensive and detailed discussion of PIR protocols appears in [11].

Next, we quickly explore some other mechanisms relying on cryptographic techniques. An approach to conceal users interests in recommendation systems is [12, 13], which propose a method that enables a community of users to calculate a public aggregate of their profiles without revealing them on an individual basis. In particular, the authors use a homomorphic encryption scheme and a P2P communication protocol for the recommender to perform this calculation. Once the aggregated profile is computed, the system sends it to users, who finally use local computation to obtain personalized recommendations. This proposal prevents the system or any external attacker from ascertaining the individual user profiles. However, its main handicap is assuming that an acceptable number of users is online and willing to participate in the protocol. In line with this, [14] uses a variant of Pailliers' homomorphic cryptosystem which improves the efficiency in the communication protocol. Another solution [15] presents an algorithm aimed at providing more efficiency by using the scalar product protocol.

2.2.3 TTP-based Mechanisms

A conceptually-simple approach to protect user privacy consists in a TTP acting as an intermediary or *anonymizer* between the user and the untrusted personalized information system. In this scenario, the system cannot know the user ID, but merely the identity of the TTP itself involved in the communication. One of the deficiencies of this approach is that personalized services cannot be provided, as the TTP forwards user data, e.g., queries, tags or ratings, of multiple users on their behalf.

As a solution to this problem, the TTP may act as a *pseudonymizer* by supplying a pseudonym ID' to the service provider, but only the TTP knows the

correspondence between the pseudonym ID' and the actual user ID. A convenient twist to this approach is the use of digital credentials [16–18] granted by a trusted authority, namely digital content proving that a user has sufficient privileges to carry out a particular transaction without completely revealing their identity. The main advantage is that the TTP need not be online at the time of service access to allow users to access a service with a certain degree of anonymity.

Unfortunately, none of these approaches prevent the service provider from profiling a user and inferring their real identity. In its simplest form, reidentification is possible due to the personally identifiable information often included in user-generated data such as Web search queries or tags. However, even though no identifying information is included, an observed user profile might be so uncommon that the attacker could narrow their focus to concentrate on a tractable list of potential identities and eventually unveil the actual user ID.

In addition to these vulnerabilities, we would like to note that a collusion of the TTP, the network operator or some entity involved in the communication could definitely jeopardize user privacy. Moreover, all TTP-based solutions require that users shift their trust from the personalized information system to another party, possibly capable of collecting user data from different applications, which finally might facilitate user profiling via cross-referencing inferences. In the end, traffic bottlenecks are a potential issue with TTP solutions.

We have shown that anonymizers, pseudonymizers and digital credentials are TTP-based approaches that may be used as an alternative to hide users' identities from an untrusted service provider. In the remainder of this subsection, we shall explore a particularly rich class of PETs that also rely on trusted entities, but whose fundamental aim is to conceal the correspondence between users exchanging messages. In the scenario of personalized information systems, *anonymous-communication systems* (ACSs) may contribute to protect user privacy against the intermediary entities enabling the communications between systems providers and users. As we shall see next, the majority of these systems build on the assumptions of the trusted model defined in Sect. 2.1. Only those systems consisting in a network of mixes may be classified into our semi-trusted model.

As commented at the beginning of Sect. 2.2, the inherent operation of the Internet poses serious privacy concerns. This is because users' IP addresses are attached to every message sent through the network. Clearly, the use of encryption techniques is not enough to mitigate such privacy risks. Hiding the content of messages hinders adversaries in their efforts to learn the information users exchange, but does not prevent those adversaries from unveiling who is communicating with whom, when, or how frequently. Motivated by this, the first high-latency ACS, Chaum's *mix* [19], appeared.

Fundamentally, a mix is a system that takes a number of input messages, and outputs them in such a way that it is infeasible to link an output to its corresponding input with certainty. In order to achieve this goal, the mix changes the appearance (by encrypting and padding messages) and the flow of messages (by delaying and reordering them). Specifically, users wishing to submit messages to other peers encrypt the intended recipients' addresses by using public key cryptography and send

these messages to the mix. The mix collects a number of these encrypted messages and stores them in its internal memory. Afterwards, these messages are decrypted and the information about senders is removed. In a last stage, when the number of messages kept reaches a certain threshold, the mix forwards *all* these messages to their recipients in a random order.

In the literature, this process of collecting, storing and forwarding messages when a condition is satisfied is normally referred to as a *round*. An important group of mixes called *pool* mixes operate on this basis. Depending on the *flushing* condition, we may distinguish different types of pool mixes. Possibly, the most relevant form of pool mixes are *threshold* pool mixes [20], where the condition is imposed on the number of messages stored, as in the case of Chaum's mixes. The main difference is that threshold pool mixes do not flush all messages in each round, but keep some of them. Clearly, this strategy degrades the usability of the system: any incoming message can be stored in the mix for an arbitrarily long period of time. But these systems, in principle, achieve a better anonymity protection since they increase the set of possible incoming messages linkable to an outgoing target message to include all those messages that entered the mix before this target message was flushed.

Another important group of pool mixes outputs messages based on time [21]. Essentially, these *timed* mixes forward all messages kept in the memory every fixed interval of time called timeout. The major advantage of these mixes is that the delay experienced by messages is upper bounded, in contrast to the case of threshold pool mixes. The flip side is that the unlinkability between incoming and outgoing messages may be seriously compromised when the number of messages arriving in that interval of time is small. Motivated by this, some of the current mix designs implement a combination of the strategies based on threshold and those based on time. Namely, these systems flush messages when a timeout expires, provided that the number of messages stored meets a threshold [22].

An alternative to pool mixes are the mixes based on the concept of *stop-and-go*, known as *continuous* mixes [23]. Specifically, this approach abandons the idea of rounds and gives the user the possibility of specifying the time that their messages will be stored in the mix before being submitted, for example, to a personalized information system. To this end, for each message to be sent the sender selects a random delay from an exponential distribution. This information is then attached to the message, which is encrypted with the mix's public key and then sent to the mix. Once the mix decrypts the message, the mix keeps it for the time specified by the user and then forwards it to its intended recipient.

The use of networks of mixes has also been thoroughly studied in the literature. The main reason to route over multiple mixes is to limit the trust that is placed on each single mix. This alternative is therefore in line with the semi-trusted model contemplated in Sect. 2.1. In order to trace messages, an adversary must ideally compromise all the mixes along the path. Depending on the network topology, we may classify the existent approaches into *cascade mixes*, *free-route networks* and *restricted-route networks*. The application of cascade mixes was already suggested by Chaum in his original work [19]. Fundamentally, this approach contemplates the concatenation of mixes to distribute trust. In contrast to this approach where messages

are routed through a fixed path, free-route networks recommend that users choose random paths to route their own messages [24]. In the end, restricted-route networks consider the case where every mix in the network is connected to a reduced number of neighboring mixes [25].

2.2.4 User Collaboration

In this subsection we examine those approaches where users collaborate to enhance their privacy. All these approaches may be understood under the semi-trusted model described in Sect. 2.1.

An archetypical example of user collaboration is the Crowds protocol [26]. This protocol is particularly helpful to minimize requirements for infrastructure and trusted intermediaries such as pseudonymizers, or to simply provide an additional layer of anonymity. In the Crowds protocol, a group of users collaborate to submit their messages to a Web server, from whose standpoint they wish to remain completely anonymous. In simple terms, the protocol works as follows. When sending a message, a user flips a biased coin to decide whether to submit it directly to the recipient, or to send it to another user, who will then repeat the randomized decision.

Crowds provides anonymity from the perspective of not only the final recipient, but also the intermediate nodes. Therefore, trust assumptions are essentially limited to fulfillment of the protocol. The original proposal suggests adding an initial forwarding step, which substantially increases the uncertainty of the first sender from the point of view of the final receiver, at the cost of an additional hop. As in most ACSs, Crowds enhances user anonymity but at the expense of traffic overhead and delay.

Closely inspired by Crowds, [27] proposes a protocol that enables users to report traffic violations anonymously in vehicular ad hoc networks. This protocol differs from the original Crowds in that, first, it does take into account transmission losses, and secondly, it is specifically conceived for multi-hop vehicular networks, rather than for wired networks. Also in the case of lossy networks, [28] provides a mathematical model of a Crowds-like protocol for anonymous communications. The authors establish quantifiable metrics of anonymity and quality of service, and characterize the trade-off between them.

Another protocol for enhancing privacy in communications, also relying on user collaboration and message forwarding, is [29]. The objective of the cited work is to hide the relationship between user identities and query contents even from the intended recipient, an information provider. The main difference with respect to the Crowds protocol is that instead of resorting to probabilistic routing with uncertain path length, it proposes adding a few forged queries.

In the context of personalized Web search, [30] proposes a P2P protocol to safeguard the privacy of users querying the Web search engine. The protocol follows the same philosophy of Crowds but leverages on social networks for grouping users with similar interests. Another approach exploiting user collaboration is [31], which suggests that two or more users exchange a portion of their queries before submitting them, in order to obfuscate their respective interest profiles versus the network

operator or external observers. The idea of query profile obfuscation through multiple user collaboration has also been investigated from a game-theoretic perspective [32].

2.2.5 Data Perturbation

An alternative to hinder an attacker in its efforts to precisely profile users consists in perturbing the information they explicitly or implicitly disclose when communicating with a personalized information system. The submission of false data, together with the user's genuine data, is an illustrative example of data-perturbative mechanism. In this kind of mechanisms, the perturbation itself typically takes place on the user side. This means that users need not trust any external entity such as the recommender, the ISP or their neighboring peers. Obviously, this does not signify that data perturbation cannot be used in combination with other TTP-based approaches or mechanisms relying on user collaboration. It is rather the opposite—depending on the trust model assumed by users, this class of PETs can be synergically combined with any of the approaches examined in Sect. 2.2. In any case, data-perturbative techniques come at the cost of system functionality and data utility, which poses a trade-off between these aspects and privacy protection.

An interesting approach to provide a distorted version of a user's profile of interests is query forgery. The underlying idea boils down to accompanying original queries or query keywords with bogus ones. By adopting this data-perturbative strategy, users prevent privacy attackers from profiling them accurately based on their queries, without having to trust neither the service provider nor the network operator, but clearly at the cost of traffic overhead. In other words, inherent to query forgery is the existence of a trade-off between privacy and additional traffic. Precisely, [33] studies how to optimize the introduction of forged queries in the setting of information retrieval.

Other alternatives relying on the principle of query forgery are [34–37], which propose a system for private Web browsing called PRAW. The purpose of this system is to preserve the privacy of a group of users sharing an access point to the Web while surfing the Internet. In order to enhance user privacy, the authors propose hiding the actual user profile by generating fake transactions, i.e., accesses to a Web page to hinder eavesdroppers in their efforts to profile the group. The PRAW system assumes that users are identified, i.e., they are logged in a Web site. However, the generation of false transactions prevents privacy attackers from the exact inference of user profiles.

The idea behind [38] is the same as in the PRAW system—the authors come up with the injection of false queries. In particular, they suggest a model working as a black box, switching between real queries and false queries. The proposed model operates as follows: it sends a real query with a certain probability, and a dummy query with the complement of that probability. The actual status of the switch and the probability of switching are assumed to be invisible or unknown to the attacker. The authors justify this assumption by arguing that this information is only available on the user side.

A software implementation of query forgery is the Web browser add-on TrackMeNot [39]. This popular add-on makes use of several strategies for generating and submitting false queries. Basically, it exploits RSS feeds and other sources of information to extract keywords, which are then used to generate false queries. The add-on gives users the option to choose how to forward such queries. In particular, a user may send bursts of bogus queries, thus mimicking the way people search, or may submit them at predefined intervals of time. Despite the strategies users have at their disposal, TrackMeNot is vulnerable to a number of attacks that leverage on the semantics of these false queries as well as timing information, to distinguish them from the genuine queries [40].

GooPIR [41] is another proposal aimed at obfuscating query profiles. Implemented as a software program,[2] this approach enables users to conceal their search keywords by adding some false keywords. To illustrate how this approach works, consider a user wishing to submit the keyword "depression" to Google and willing to send it together with two false keywords. Based on this information, GooPIR would check the popularity of the original keyword and find that "iPhone" and "elections" have a similar frequency of use. Then, instead of submitting each of these three keywords at different time intervals, this approach would send them in a batch. The proposed strategy certainly thwarts attacks based on timing. However, its main limitation is that it cannot prevent an attacker from combining several of these batches, establishing correlations between keywords, and eventually inferring the user's real interest [42]. As an example, suppose that the user's next query is "prozac" and that GooPIR recommends submitting it together with the keywords "shirt" and "eclipse". In this case, one could easily deduce that the user is interested in health-related issues.

Naturally, the perturbation of user profiles for privacy preservation may be carried out not only by means of the insertion of bogus activity, but also by *suppression*. An example of this latter kind of perturbation may be found in [43], where the authors propose the elimination of tags as a privacy-enhancing strategy in the scenario of the semantic Web. On the one hand, this strategy allows users to enhance their privacy to a certain degree, but on the other it comes at the cost of a degradation in the semantic functionality of the Web, as tags have the purpose of associating meaning with resources. Precisely, [44] investigates mathematically the privacy-utility trade-off posed by the suppression of tags, measuring privacy as the Shannon's entropy of the perturbed profile and utility as the percentage of tags users are willing to eliminate. Intimately related to this work is [45], where the impact of tag suppression is assessed experimentally in the context of resource recommendation and parental control, in terms of percentages regarding missing tags on resources on the one hand, and in terms of false positives and negatives on the other.

The combined use of both strategies, that is, forgery and suppression, is studied in the scenario of personalized recommendation systems [46]. With the adoption of those strategies, users may wish to submit false ratings to items that do not reflect their preferences, and/or refrain from rating certain items they have an opinion on. The trade-off posed by these perturbative strategies in terms of privacy protection

[2] http://unescoprivacychair.urv.cat/goopir.php.

Table 1 Summary of the most relevant privacy-preserving approaches in terms of the trust model and technology assumed

Approaches	Underlying mechanism	Trust model	Disadvantages
PIR [9, 10]	Cryptographic methods	Untrusted	No personalization
			Database owner must collaborate
			Computational overhead
Anonymizer	TTP	Trusted	Users must trust an external entity
Pseudonymizer			Vulnerable to collusion attacks
Digital credentials [16–18]			Traffic bottlenecks
Mix-based systems [19–23, 49]	TTP	Trusted	Delay experienced by messages
			Users must trust an external entity
			Vulnerable to collusion attacks
			Infrastructure requirements
Crowds and other P2P protocols [26–32]	User collaboration	Semi-trusted	Numerous users must collaborate
			Vulnerable to collusion attacks
			Traffic overhead
Query forgery [33–39]	Data perturbation	Untrusted	Traffic overhead
Tag suppression [43–45]	Data perturbation	Untrusted	Semantic loss incurred by suppressing tags

and data utility is investigated analytically in [47]. The authors find a closed-form solution to the problem of optimal simultaneous forgery and suppression of ratings, and evaluate their approach in the real-world recommender Movielens.

Lastly, another form of perturbation [48] consists in hiding certain categories of interests. In this work, user profiles are organized in a hierarchy of categories in such a way that lower-levels categories are regarded as more specific than those at higher levels. Based on this user-profile model, the idea is to disclose only those parts of the user profile corresponding to high-level interests. Table 1 summarizes the major conclusions of this section.

3 Privacy Metrics

As discussed in Sect. 1, personalized information systems rely on some form of profiling to provide information tailored to users' preferences. Said otherwise, personalization comes at the risk of profiling. The literature of privacy metrics in this particular scenario typically measures user privacy based on the profile constructed by an attacker. Potential privacy attackers include the systems themselves but also any other entity capable of eavesdropping the information users reveal to such systems. As we shall see next, most of the proposed metrics quantify user privacy according to two profiles. The former is the profile capturing the genuine interests of a user, and the latter the profile observed by the attacker. In principle, the observed profile does not need to coincide with the original one. This may be as a result of adopting any of the PETs reviewed in Sect. 2.2. Despite the variety of PETs examined in that section, the vast majority of privacy metrics in the context of personalized information systems are specifically conceived to evaluate data-perturbative mechanisms, collaborative techniques and ACSs. Next, we review some of the most relevant metrics for these three important classes of PETs.

In the setting of personalized Web search, [34] proposes PRAW, a system aimed at preserving the privacy of a group of users sharing an access point to the Web. The cited work and its successive improvements [35–37, 50, 51] suggest perturbing the actual user profile by generating fake transactions, that is, accesses to Web pages. In the PRAW system, user profiles are modeled as weighted vectors of queries, and privacy is computed as the similarity between the genuine profile and that observed from the outside. More specifically, the authors use the cosine measure [52] to capture the similarity between both profiles. They assume, accordingly, that the lower the cosine similarity value between these two profiles, the higher the privacy level attained by such perturbation strategy.

Similarly to those works, [53] proposes to measure privacy as a generic function of both the actual profile and the profile observed by a recommender. The authors acknowledge that this function may, in principle, be different for each user, as users may perceive privacy risks differently. Their metric is justified in the same way as in the PRAW system. That is, it is assumed that the more those profiles differ, the higher the privacy protection. Then, a weighted version of the Euclidean distance is given as a particular instantiation of the generic function. The main problem with PRAW and this latter approach is that neither justifies the choice of the similarity and distance functions, neglecting alternatives such as the Pearson and Jaccard correlation coefficients, or any Minkowski distance.

In the literature we also find examples of privacy criteria based on information-theoretic quantities. In the context of personalized Web search, for example, [38] identifies two privacy breaches when submitting search queries. The former refers to the disclosure of identifying information, e.g., asking Google Maps how to get from your home to a restaurant. The latter refers to private information inferred indirectly from such queries, e.g., estimating the probability of suffering from a disease based on searches for medical assistance. The authors propose the injection of false queries

to counter the latter kind of privacy breach, and quantify privacy as the mutual information between the real queries X and the observed ones Y. Recall [54] that the mutual information between two random variables (r.v.'s) may be interpreted as a measure of their mutual dependence. Accordingly, when the mutual information is zero, the authors argue that the observed profile does not leak any information about the actual profile, and thus perfect privacy protection is attained.

Still in the scenario of personalized Web search, [30] defines a privacy criterion called *profile exposure level*. This criterion uses the mutual information between the genuine queries of a given user and the queries submitted to the search engines, including the genuine ones and those forwarded by this user on behalf of their neighbors. Specifically, user privacy is measured as the quotient between the mutual information and the Shannon entropy[3] of the distribution of original queries. In the end, the authors justify their metric by interpreting it as an amount of uncertainty reduction [54]. Another metric for a privacy-enhancing collaborative mechanism is proposed in [27]. In particular, the cited work proposes a variation of the Crowds protocol for vehicular ad hoc networks, and measures user anonymity as the attacker's probability of error when guessing the identity of the sender of a given message, in keeping with [55].

Another information- theoretic privacy criterion is [48]. In this approach, user profiles are represented essentially as normalized histograms of queries. The profile categories are organized hierarchically so that the higher-level interests are more general than those at the lower levels. According to this representation, the authors define user privacy based on two parameters, *minDetail* and *expRatio*. The former parameter is a threshold that is used to filter out those components of the profile where the user has shown little interest in. The latter is the Shannon entropy of the filtered profile, a quantity that is taken as the level of privacy achieved.

In all these information-theoretic metrics, the justification consists merely in noting that entropy is a measure of uncertainty and mutual information is a measure of the reduction in uncertainty. While there is some intuition behind these criteria, the authors do not justify the choice, ignoring other measures of uncertainty, for example, from the field of information theory. Besides, these metrics are often not defined in terms of an adversary model that contemplates assumptions such as the attacker's capabilities or objectives. Ultimately, they are conceived specifically for assessing the effectiveness of concrete privacy-preserving mechanisms.

An information-theoretic measure of privacy that is rigorously justified and that is not tied to any particular privacy-enhancing mechanism is [56–58]. The proposed metric is the Kullback-Leibler (KL) divergence [54], a quantify that, although it is not a distance function, it does provide a measure of discrepancy between distributions. The KL divergence is often referred to as *relative entropy*, as it may be regarded as a generalization of the Shannon entropy of a distribution, relative to another.

The authors interpret both the KL divergence and Shannon's entropy under two distinct adversary models, defined consistently withthe technical literature of

[3] Shannon's entropy of a discrete r.v. is a measure of the uncertainty of the outcome of this r.v.

profiling. First, they consider an attacker who strives to target users who deviate from the average profile of interests; and secondly, the authors contemplate an attacker whose objective is to classify a given user into a predefined group of users.

In the former model, the use of KL divergence is justified by elaborating on Jaynes' rationale behind entropy-maximization methods [59] and the method of types [54], Sect. 11 of large deviation theory. In essence, this justification builds on three main principles. First, the authors model the profile of a user as a type or empirical distribution. Secondly, through Jaynes' rationale, the KL divergence between the user's profile and the population's is deemed as a measure of the probability of the former profile. And thirdly, they consider that the probability of a profile may be a suitable measure of its anonymity. Only under this interpretation, the uniform profile is of particular interest since entropy may be justified as anonymity criterion in a sense entirely analogous to that of divergence.

In the latter adversary model, the authors propose measuring privacy as the KL divergence between the user's apparent profile and the distribution of the group this user does not want to be classified into. The authors interpret this privacy criterion as false positives and negatives when an attacker applies a binary hypothesis test to find out whether a sequence of observed data belongs to the sensitive group or not. If the distribution of this group is unavailable to the user, their actual profile is assumed to be the group's. Under this assumption, the user's strategy consists in maximizing the discrepancy between the apparent profile and their genuine profile. Conceptually, this reflects the situation in which a user does not want the perturbed, observed profile resemble their actual profile. This is in line with the assumptions of the similarity-based criteria examined above.

Having examined some of the most relevant privacy metrics for data-perturbative mechanisms and collaborative technologies, next we explore several anonymity measures amply utilized in the field of ACSs.

In the important case of the mix systems reviewed in Sect. 2.2, [23] defined the *anonymity set* of users as the set of possible senders of a given message, or recipients, in the sense that the likelihood of them fulfilling the role in question is nonzero. A simple measure of anonymity was proposed by [60], namely the logarithm of the number of users involved in the communication, that is, the Hartley entropy of the anonymity set. The main drawback of this metric is that it does not contemplate the probabilistic information that an adversary may obtain about users when observing the system. In other words, this approach ignores the fact that certain users may be more likely to be the senders of a particular message.

Several approaches have considered the use of information-theoretic quantities to evaluate ACSs. The most significant are those proposed in [61, 62], in which the degree of anonymity observable by an adversary is measured essentially as the Shannon entropy of the probability distribution of possible senders of a given message. A well-known interpretation of Shannon's entropy refers to the game of 20 questions, in which one player must guess what the other is thinking through a series of yes/no questions, as quickly as possible. Informally, Shannon's entropy is a lower bound on—and often good approximation to the minimum of—the average number of binary questions regarding the nature of possible outcomes of an event, to

determine which one in fact has come to pass, intelligently exploiting their known probabilities.

Still in the case of information-theoretic measures, [63] formalizes the notion of unlinkability by using Shannon's entropy. By contrast, [64, 65] argue that a worst-case metric should be considered instead of Shannon's entropy, since the latter contemplates an average case. The authors refer to this worst-case metric as *local anonymity*, essentially equivalent to min-entropy, and concordantly define the *source hiding* property as the requirement that no sender probability exceed a given threshold. Another approach [66] proposes a method for quantifying the property of *relationship anonymity*, as defined in [67]. More specifically, the authors make use of Shannon's entropy and min-entropy for measuring this property. Similarly, [68] evaluates Shannon's entropy, min-entropy and Hartley's entropy as anonymity metrics, and proposes then to use Rényi's entropy, which may be regarded as a generalization of those three metrics.

Lastly, [69] tackles the problem of designing threshold pool mixes in a manner that contemplates the optimal trade-off between user anonymity and delay. The authors approach this problem by adopting several quantifiable measures of anonymity in the literature, Hartley's entropy, Shannon's entropy, min-entropy, and collision entropy.

4 Conclusions

In recent times we are witnessing the emergence of a new generation of information systems that adapt their functionalities to meet the unique needs of each individual. Personalization is revolutionizing the manner we access information but, at the same time, it is raising new privacy concerns with respect to user profiling.

In this paper, we started by reviewing some of the most relevant privacy-enhancing mechanisms in the scenario of personalized information systems. To this end, we classified such mechanisms into five main groups: mechanisms which prevents users from being tracked; cryptography-based methods from PIR; technologies that build on TTP such as anonymizers, pseudonymizers and ACSs; approaches relying on the principle of user collaboration; and techniques that perturb user's private data.

Then, we surveyed the literature of privacy metrics in this scenario, with a special emphasis on those specifically intended for data-perturbative techniques. We showed that most of the criteria for quantifying the privacy of user profiles reduce to functions that take as inputs the actual user profile and the profile observed from the outside. Our survey classified these criteria into similarity-based privacy measures and uncertainty-based privacy metrics, and concluded that most of them are merely ad hoc proposals for specific applications and, what is more important, are not appropriately justified. This undoubtedly indicates that the problem of quantifying user privacy is still in its infancy and that a vast space of unexplored models remain to be discovered.

References

1. Ritholtz, B.: Things that happen on internet every sixty seconds. http://www.ritholtz.com/blog/2011/12/60-seconds-things-that-happen-every-sixty-seconds/ (2011)
2. Grossman, W.M.: alt.scientology.war (1996)
3. AOL search data scandal: http://en.wikipedia.org/wiki/AOL-search-data-leak (2006). Accessed 15 Nov 2013
4. Shen, X., Tan, B., Zhai, C.: Privacy protection in personalized search. ACM Spec. Interest Group Inform. Retrieval (SIGIR). Forum **41**(1), 4–17 (2007)
5. Srisuresh, P., Holdrege, M.: IP network address translator (NAT) terminology and considerations. RFC 2663 (Informational) (1999)
6. Droms, R.: Dynamic host configuration protocol. RFC 2131 (Draft Standard) Updated by RFCs 3396, 4361, 5494, 6842 (1997)
7. Ostrovsky, R., Skeith III, W.E.: A survey of single-database PIR: Techniques and applications. In: Proceedings of International Conference on Practice, Theory Public-Key Cryptography (PKC). Lecture Notes in Computer Science (LNCS), vol. 4450, pp. 393–411. Beijing, China, Springer-Verlag (2007)
8. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.L.: Private queries in location based services: anonymizers are not necessary. In: Proceedings ACM SIGMOD International Conference Management of Data, pp. 121–132. Vancouver, Canada (2008)
9. Chor, B., Goldreich, O., Kushilevitz, E., Sudan, M.: Private information retrieval. In: Proceedings IEEE Annual Symposium Foundations of Computer Science (FOCS), pp. 41–50. Milwaukee, WI (1995)
10. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: single database, computationally-private information retrieval. In: Proceedings of the IEEE Annual Symposium Foundations on Computer Science (FOCS), pp. 364–373. IEEE Computer Society (1997)
11. Yekhanin, S.: Private information retrieval. Commun. ACM **53**(4), 68–73 (2010)
12. Canny, J.: Collaborative filtering with privacy via factor analysis. In: Proceedings ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 238–245. Tampere, Finland, ACM (2002)
13. Canny, J.F.: Collaborative filtering with privacy. In: Proceedings of the IEEE Symposium on Security and Privacy (SP), pp. 45–57 (2002)
14. Ahmad, W., Khokhar, A.: An architecture for privacy preserving collaborative filtering on Web portals. In: Proceedings IEEE International Symposium on Information Assurance and Security (IAS), pp. 273–278. IEEE Computer Society, Washington, DC (2007)
15. Zhan, J., Hsieh, C.L., Wang, I.C., Hsu, T.S., Liau, C.J., Wang, D.W.: Privacy-preserving collaborative recommender systems. IEEE Trans. Syst. Man, Cybern. **40**(4), 472–476 (2010)
16. Chaum, D.: Security without identification: transaction systems to make big brother obsolete. Commun. ACM **28**(10), 1030–1044 (1985)
17. Benjumea, V., López, J., Linero, J.M.T.: Specification of a framework for the anonymous use of privileges. Telemat. Informat. **23**(3), 179–195 (2006)
18. Bianchi, G., Bonola, M., Falletta, V., Proto, F.S., Teofili, S.: The SPARTA pseudonym and authorization system. Sci. Comput. Program. **74**(1–2), 23–33 (2008)
19. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudonyms. Commun. ACM **24**(2), 84–88 (1981)
20. Serjantov, A., Dingledine, R., Syverson, P.: From a trickle to a flood: active attacks on several mix types. In: Proceedings of Information Hiding Workshop (IH), pp. 36–52. Springer-Verlag (2002)
21. Serjantov, A., Newman, R.E.: On the anonymity of timed pool mixes. In: Proceedings of the Workshop on Privacy and Anonymity Issues in Networked and Distributed Systems, pp. 427–434. Kluwer (2003)
22. Möller, U., Cottrell, L., Palfrader, P., Sassaman, L.: Mixmaster protocol—version 2. Internet draft, Internet Eng. Task Force (2003) Accessed 18 Feb 2014.

23. Kesdogan, D., Egner, J., Büschkes, R.: Stop-and-go mixes: providing probabilistic anonymity in an open system. In: Proceedings of Information Hiding Workshop (IH), pp. 83–98. Springer-Verlag (1998)
24. Rennhard, M., Plattner, B.: Practical anonymity for the masses with mix-networks. In: Proceedings of International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), pp. 255–260. IEEE Computer Society (2003)
25. Danezis, G.: Mix-networks with restricted routes. In: Proceedings of International Symposium on Privacy Enhancing Technologies Symposium (PETS). Lecture Notes in Computer Science (LNCS), pp. 1–17 (2003)
26. Reiter, M.K., Rubin, A.D.: Crowds: anonymity for Web transactions. ACM Trans. Inform. Syst. Secur. **1**(1), 66–92 (1998)
27. Tripp-Barba, C., Urquiza, L., Aguilar, M., Parra-Arnau, J., Rebollo-Monedero, D., J. Forné, E.P.: A collaborative protocol for anonymous reporting in vehicular adhoc networks. Comput. Stand. Interf. **36**:1, 188–197 (2013) (To appear)
28. Rebollo-Monedero, D., Forné, J., Pallarès, E., Parra-Arnau, J., Tripp, C., Urquiza, L., Aguilar, M.: On collaborative anonymous communications in lossy networks. Secur Commun. Netw (2013). doi:10.1002/sec.793
29. Rebollo-Monedero, D., Forné, J., Solanas, A., Martnez-Ballesté, T.: Private location-based information retrieval through user collaboration. Comput. Commun. **33**(6), 762–774 (2010)
30. Erola, A., Castellà-Roca, J., Viejo, A., Mateo-Sanz, J.M.: Exploiting social networks to provide privacy in personalized Web search. J. Syst. Softw. **84**(10), 1734–745 (2011)
31. Rebollo-Monedero, D., Forné, J., Domingo-Ferrer, J.: Coprivate query profile obfuscation by means of optimal query exchange between users. IEEE Trans. Depend. Secure Comput. **9**(5), 641–654 (2012)
32. Domingo-Ferrer, J., González-Nicolás, Ú.: Rational behavior in peer-to-peer profile obfuscation for anonymous keyword search. Inform. Sci. **185**(1), 191–204 (2012)
33. Rebollo-Monedero, D., Forné, J.: Optimal query forgery for private information retrieval. IEEE Trans. Inform. Theory **56**(9), 4631–4642 (2010)
34. Elovici, Y., Shapira, B., Maschiach, A.: A new privacy model for hiding group interests while accessing the Web. In: Proceedings of Workshops on Privacy in the Electronic Society, pp. 63–70. ACM, Washington, DC (2002)
35. Elovici, Y., Shapira, B., Maschiach, A.: A new privacy model for Web surfing. In: Proceedings of International Workshop on Next Generation Information Technologies and System (NGITS), pp. 45–57.Springer-Verlag (2002)
36. Elovici, Y., Glezer, C., Shapira, B.: Enhancing customer privacy while searching for products and services on the World Wide Web. Internet Res. **15**(4), 378–399 (2005)
37. Elovici, Y., Shapira, B., Meshiach, A.: Cluster-analysis attack against a private Web solution (PRAW). Online Inform. Rev. **30**, 624–643 (2006)
38. Ye, S., Wu, F., Pandey, R., Chen, H.: Noise injection for search privacy protection. In: Proceedings of IEEE International Conference on Computational Science and Engineering, pp. 1–8. IEEE Computer Society (2009)
39. Howe, D.C., Nissenbaum, H.: TrackMeNot: Resisting surveillance in Web search. In: Lessons from the Identity Trail: Privacy, Anonymity and Identity in a Networked Society, pp. 417–436. Oxford University Press, NY (2009)
40. Chow, R., Golle, P.: Faking contextual data for fun, profit, and privacy. In: Proceedings of ACM workshop on Privacy in the Electronic Society, pp. 105–108. ACM (2009)
41. Domingo-Ferrer, J., Solanas, A., Castellà-Roca, J.: $h(k)$-private information retrieval from privacy-uncooperative queryable databases. Online Inform. Rev. **33**(4), 720–744 (2009)
42. Balsa, E., Troncoso, C., Daz, C.: OB-PWS: Obfuscation-based private Web search. In: Proceedings of IEEE Symposium on Security and Privacy (SP), pp. 491–505. IEEE Computer Society (2012)
43. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: A privacy-preserving architecture for the semantic Web based on tag suppression. In: Proceedings of International Conference on Trust, Privacy and Security in Digital Business (TrustBus). Lecture Notes in Computer Science (LNCS), vol. 6264, pp. 58–68. Bilbao, Spain (2010)

44. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J., Muñoz, J.L., Esparza, O.: Optimal tag suppression for privacy protection in the semantic Web. Data Knowl. Eng. **81–82**, 46–66 (2012)
45. Parra-Arnau, J., Perego, A., Ferrari, E., Forné, J., Rebollo-Monedero, D.: Privacy-preserving enhanced collaborative tagging. IEEE Trans. Knowl. Data Eng. **26**(1), 180–193 (2014)
46. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: A privacy-protecting architecture for collaborative filtering via forgery and suppression of ratings. In: Proceedings of the International Workshop on Data Privacy Management (DPM). Lecture Notes in Computer Science (LNCS), vol. 7122, pp. 42–57. Leuven, Belgium (2011)
47. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: Optimal forgery and suppression of ratings for privacy enhancement in recommendation systems. Entropy **16**(3), 1586–1631 (2014)
48. Xu, Y., Wang, K., Zhang, B., Chen, Z.: Privacy-enhancing personalized web search. In: Proceedings of the International WWW Conference, pp. 591–600. ACM (2007)
49. Goldschlag, D., Reed, M., Syverson, P.: Hiding routing information. In: Proceedings of International Workshop on Information Hiding (IH), pp. 137–150 (1996)
50. Kuflik, T., Shapira, B., Elovici, Y., Maschiach, A.: Privacy preservation improvement by learning optimal profile generation rate. In: User Modeling. Lecture Notes in Computer Science (LNCS), vol. 2702, pp. 168–177. Springer-Verlag (2003)
51. Shapira, B., Elovici, Y., Meshiach, A., Kuflik, T.: PRAW—The model for PRivAte Web. J. Am. Soc. Inform. Sci. Technol. **56**(2), 159–172 (2005)
52. Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., Stum, G.: Evaluating similarity measures for emergent semantics of social tagging. In: Proceedings of the International WWW Conference, pp. 641–650. ACM (2009)
53. Halkidi, M., Koutsopoulos, I.: A game theoretic framework for data privacy preservation in recommender systems. In: Proceedings of European Machine Learning Principles and Practice of Knowledge Discovery in Databases (ECML PKDD), pp. 629–644. Springer-Verlag (2011)
54. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley, New York (2006)
55. Rebollo-Monedero, D., Parra-Arnau, J., Diaz, C., Forné, J.: On the measurement of privacy as an attacker's estimation error. Int. J. Inform. Secur. **12**(2), 129–149 (2012)
56. Parra-Arnau, J., Rebollo-Monedero, D., Forné, J.: Measuring the privacy of user profiles in personalized information systems. Future Gen. Comput. Syst. (FGCS), Special Issue Data Knowl. Eng. **33**, 53–63 (2014)
57. Rebollo-Monedero, D., Parra-Arnau, J., Forné, J.: An information-theoretic privacy criterion for query forgery in information retrieval. In: Proceedings of International Conference on Security Technology (SecTech). Communications in Computer and Information Science (CCIS), vol. 259, pp. 146–154. Jeju Island, South Korea, Springer-Verlag (2011)
58. Parra-Arnau, J.: Privacy protection of user profiles in personalized information systems. PhD Thesis, Technical University Catalonia (UPC) (2013)
59. Jaynes, E.T.: On the rationale of maximum-entropy methods. Proc. IEEE **70**(9), 939–952 (1982)
60. Berthold, O., Pfitzmann, A., Standtke, R.: The disadvantages of free MIX routes and how to overcome them. In: Proceedings of Design. Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity Unobser. Lecture Notes in Computer Science (LNCS), pp. 30–45. Berkeley, CA, Springer-Verlag (July 2000)
61. Díaz, C., Seys, S., Claessens, J., Preneel, B.: Towards measuring anonymity. In: Proceedings of International Symposium on Privacy Enhancing Technologies (PETS). Lecture Notes in Computer Science (LNCS), vol. 2482, pp. 54–68. Springer-Verlag (2002)
62. Serjantov, A., Danezis, G.: Towards an information theoretic metric for anonymity. In: Proceedings of International Symposium on Privacy Enhancing Technologies (PETS), vol. 2482, pp. 41–53. Springer-Verlag (2002)
63. Steinbrecher, S., Kopsell, S.: Modelling unlinkability. IIn: Proceedings of International Symposium on Privacy Enhancing Technologies (PETS), pp. 32–47. Springer-Verlag (2003)
64. Tóth, G., Hornák, Z., Vajda, F.: Measuring anonymity revisited. In: Proceedings of Nordic Workshop Secure IT Systems, pp. 85–90 (2004)
65. Tóth, G., Hornák, Z.: Measuring anonymity in a non-adaptive, real-time system. In: Proceedings of International Symposium on Privacy Enhancing Technologies (PETS). Lecture Notes in Computer Science (LNCS), vol. 3424, pp. 226-241. Toronto, Canada, Springer-Verlag (2004)

66. Shmatikov, V., Wang, M.H.: Measuring relationship anonymity in mix networks. In: Proceedings of Workshops on Privacy in the Electronic Society, pp. 59–62. ACM (2006)
67. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. v 0.34 (2010)
68. Clauß, S., Schiffner, S.: Structuring anonymity metrics. In: Proceedings of ACM Workshop on Digital Identity Management, pp. 55–62. Fairfax, VA, ACM (2006)
69. Rebollo-Monedero, D., Parra-Arnau, J., Forné, J., Diaz, C.: Optimizing the design parameters of threshold pool mixes for anonymity and delay. Comput. Netw. (2014) (To appear)

Managing Privacy in the Internet of Things: DocCloud, a Use Case

Juan Vera del Campo, Josep Pegueroles, Juan Hernández Serrano and Miguel Soriano

Abstract In this chapter, we describe nodes in the Internet of Things can configure themselves automatically and offer personalized services to the users while protecting their privacy. We will show how privacy protection can be achieved by means of a use case. We describe DocCloud, a recommender system where users get content recommended by other users based on their personal affinities. To do this, their things connect together based on the affinities of their owners, creating a social network of similar things, and then provide the recommender system on top of this network. We present the architecture of DocCloud and analyze the security mechanisms that the system includes. Specifically, we study the properties of plausible deniability and anonymity of the recommenders and intermediate nodes. In this way, nodes can recommend products to the customers while deny any knowledge about the product they are recommending or their participation in the recommendation process.

1 Introduction

The Internet of Things paradigm let small objects to interact with other devices autonomously. The participants in the Internet of Things are able to detect changes in the environment or receive orders and data from other Things and users, and respond to these inputs in a smart way to provide new services personalized to the current context of the client.

When users -either humans or machines- join an Internet of Things, they make lots of decisions about their environment. For example, which of the Things, services and data they own must be available to the rest of the network, or which Things already in the network must be contacted. To do so, users of an Internet of Things register their devices in the network and select and use services already registered according

J.V. del Campo (✉) · J. Pegueroles · J. Hernández Serrano · M. Soriano
Universitat Politècnica de Catalunya, Barcelona, Spain
e-mail: juanvi@entel.upc.edu

G. Navarro-Arribas and V. Torra (eds.), *Advanced Research in Data Privacy*,
Studies in Computational Intelligence 567, DOI 10.1007/978-3-319-09885-2_24

to their needs. In order to decide which Thing available in the network is the most suitable entity to be contacted, they need some semantic description that captures their needs and the kind of services they offer. Since in the Internet of Things this selection is automatic, the system must include some mechanism that captures the semantic description of the users' needs and the capabilities of the devices, documents and resources shared within the network. In addition to this semantic language, the system must provide a mechanism that takes these descriptions as inputs and outputs the resources the user must access. This is a recommender system.

The process of receiving a useful recommendation begins with the creation of a description of the users that captures their interests. These are the users' profiles, and they include sensitive information, including their needs, likes and dislikes. The protection of these personal data is not only a necessity for the users; it can improve the result of the recommendation process. Indeed, if users are not afraid of declaring their likes and dislikes, the recommendations that they get from the system will be more accurate. Protecting private data is not the only security service to provide in recommender systems. In a distributed environment, other actors of the system may need additional protection. The providers of a recommendation, for example, expose their own opinion of the resource that they are recommending. Thus, recommenders should be protected in the same way than users. The risk of being exposed may affect the quality of the output of the system, for example, preventing the recommendation of a certain service even if the recommender thinks that it is the most suitable for the user.

1.1 Our Contribution

We will explore how to protect the privacy of the users in the Internet of Things by describing a use case. We will introduce DocCloud, a decentralized recommender system on a social network created using the Internet of Things paradigm. A *recommender system* is an automatic system that, given a set of available products and a model that captures the interests of a user, outputs a list of products that the system estimates will be of interest to the user. We identify five different roles in DocCloud: *merchants* that provide resources and resource descriptions, or *profiles*; *customers* that request recommendations according to their *user profiles*; *indexers* that reply to queries; *repositories* that provide access to final resources and *intermediate nodes* that route messages from merchants and customers to indexers and repositories. We will protect individual Things and their owners by means of hiding the identity of the node that outputs a recommendation, and providing mechanisms to calculate affinities without leaking all the personal information in the user profile used by the recommender system.

We will consider that any Thing in the network may be an attacker of the system and individual Things cannot be trusted. There are at least four ways for Things to learn information about the network and resources: (i) inspection of the messages they route, (ii) analysis of their links to other nodes, (iii) collusion with other nodes to get

information about other parts of the network, and (iv) effectively using the services that the network offers. Document indexers have an additional way to attack the system: (v) they can analyze the profiles they store. Since the attackers are individual Things, we will limit their power to the power needed for a fair use of the system. We define that, if an attack needs more processing power than a Thing needs to run the system, it cannot be prosecuted for not providing this power. We will build the definition of plausible deniability on this property.

DocCloud provides these security services:

- *Indexer plausible deniability* Indexers have the property of plausible deniability if it is not reasonable to force them to run the process to identify the resources they are recommending. However, indexers still should be able to provide correct recommendations. In this way, indexers are not aware of the resource profiles they are serving with some probability.
- *Oblivious routing* Intermediate Things should not be aware of the content of the messages they are routing from the point of view of plausible deniability. In this way, nodes that assist in locating resources by means of routing queries cannot be accused of abetting copyright infringement.
- *Indexer anonymity* Customers do know the query that they send to the system and the results of this query. If prosecutors acting as customers are able to identify the indexer that answers a query about a sensitive resource, they may accuse the indexer of abetting the access to the resource.

In Sect. 2, we present the architecture of our proposal. Then, Sect. 2.1 details the sequence that a user follows to obtain a recommendation from the system. Finally, Sect. 3 analyzes some of the mechanisms proposed in the other sections.

2 DocCloud System Architecture

We formalize next the steps that DocCloud takes for providing a recommendation about the most suitable Thing, service or resource is available the environment according to the needs of the users.

1. **Resources collection and profiling** During this step, the system collects and identifies the resources available to the network. These resources may be Things, services offered by them or external resources such as commercial products. This step involves the creation of a profile that captures the defining characteristics the Things and the resources and services they share.
2. **User profiling** During this step, the object owned by a user enters the network. In this moment, a user profile is created and assigned to the user. This profile could be controlled by the user, for example, if it is the output on the answers of a test or a self-configuration of her needs. The profile can also be created by an external observer by means of the analysis of the user behavior. This is the case of profiles that involve the study of the buying habits of the users. In any case, the

information that is included in the user profile is highly sensitive and the system must provide mechanisms to protect and secure this profile.

3. **Recommender selection** The network may have many entities that register and are able to select the most suitable object, service or item after a recommendation request. For example, some recommenders may be specialized only on some object categories, or being specific for some context. During this step, the system identifies and selects those recommenders that are more suitable to answer the query of the user. For example, the number of users and Things in a shopping mall may be in the order of thousands. In order to manage this amount of information, an initial classification of Things according to some criteria (proximity, affinity...) takes place. In a social network, participants often select an initial set of "friends" or "similar people" that can be used to make recommendations.
4. **Query the system** During this step, participants send a query to the system, which includes a semantic description of the object she is interested in. The complexity of the process of querying the system varies with the different recommender types. For example, this is a very simple process in a centralized repository shop, since it is reduced to sending a message to some database. In distributed systems, on the other hand, this step involves routing the query to the selected recommenders and it may be a complex task. As in the case of users' profiles, the query of a user to the Internet of Things includes sensitive information that must be protected.
5. **Recommendation process** The selected recommenders search their internal databases to select those recommenders that are more suitable to answer the query of a user. Then, the recommenders return a set of Things they believe that are interesting to the user. At this point, we find useful to imagine a recommender system as a system where users evaluate Things. In this case, we can model the knowledge of the system as a matrix, where rows are users and columns resources. This matrix is scarcely populated and most of the elements are empty, since it is usually impossible for users to evaluate a significant subset of the available resources. The goal of the recommender system is making a good guess of the rate that a user would give to a resource that is not yet evaluated. Given these guesses, the recommender decides whether a resource interests the user or not with an algorithm that is often as simple as "the resource is interesting if its calculated rate is higher than a threshold λ". The mechanisms that are used to populate the elements of the matrix, the input that the recommender needs for guessing rates and the actual location of the matrix in the system are the main differences between the different implementations of real recommender systems.
6. **Accessing the recommended resources** During the final phase of the system, users access the recommended resources. The final output of the process may be useful to enhance future recommendations, and hence some feedback mechanism can be included. This is the case of user profiles that are based on buying habits. From the security point of view, an access to a resource implies that a recommendation was correct, and since it tells something about the user profile, this is a security leakage. Even if the other steps of the process are conveniently protected, an attacker may learn something about a user's profile by means of

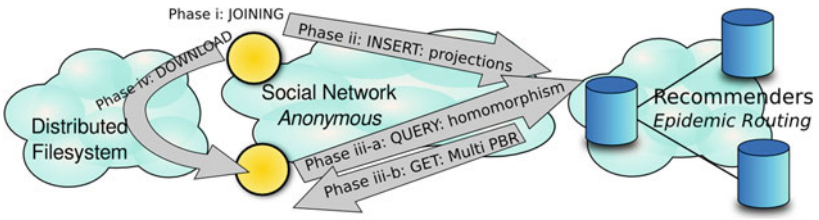


Fig. 1 An overview of the security mechanisms

inspecting only the resources the user accesses. A system that aims to protect the user's privacy must consider the final access to the resources as part of process to process.

2.1 Recommender System Operation

To simplify the description of DocCloud, we summarize the previously described phases in four steps: (i) the creation of a social cloud, (ii) the insertion of resource profiles into the indexers, (iii) the search of recommendations by the customers and (iv) access of the recommended resource. Figure 1 shows these steps and the mechanisms used in them.

2.1.1 JOINING: Creation of the Social Network

We aim to improve the performance of the Internet of Things based on the creation of a social overlay on top of an unstructured P2P network. Objects are clustered according to their affinity. For this approach to be successful, it requires fast identification and location of clusters or other users that are similar, and an efficient construction of these clusters. On this social network, other complex services based on similarities are easily deployable.

In DocCloud, we introduced a mechanism to find similar Things in an efficient and fast way [1]. This mechanism used epidemic routing. Epidemic routing is a convenient method to distribute messages when data must arrive to as many nodes with shared features as possible. From here onward, we assume there is an epidemic routing algorithm inside the different clusters of networks to distribute messages and support the discovering of other affine users. Other proposals that use epidemics routing in a way similar to ours are [2, 3].

When a customer joins a cluster of similar users, a new key must be created and distributed among the group's members. We propose the key management algorithm of Hernández-Serrano et al. [4]. This is a Group Key Management (GKM) scheme, which manages the changes of the shared key during the life of a group. The main

challenge of a GKM scheme is the secure update and distribution of the shared key among the group's members.

The output of the key management scheme produces a shared key that all members of the group know. This key is updated when a new member joins the group, or an existing member leaves. All nodes in a cluster A will agree on a secret key that includes two items, $K_A = (B, M_A)$. The first part of the key is a reference to another cluster of nodes B that will be used to store the profiles of the resources shared by nodes in A. The reader should note that nodes in B will not need to be aware of the identity of the cluster A. The second part of K_A is a random matrix M_A that we will use to protect profiles in A.

2.1.2 INSERT: Inserting Resource Profiles Into Indexers

After users join a specific cluster of the social network, they will share some resources with the rest of the network. During this phase, user $a \in A$ plays the role of *merchant*. For a resource d_i, a assigns a profile $\bar{p}(d)$ and publishes the resource d under $URL(d)$ in the social network created by the Things, as described in Sect. 2. Finally, a inserts the pair $(\bar{p}(d), URL(d))$ into a random indexer $b \in B$.

The reader will notice that if indexers and intermediate nodes are able to access these profiles in clear, they will know exactly what kind of resources they are providing access to. In addition, they can even estimate the user profile by simply collecting enough resource profiles from the same source. Indexers and intermediate nodes need to use these profiles to route and answer queries according to the affinity of the user to the resource descriptions that they index. We need to devise a mechanism that hides some of the information in the descriptions but is still useful to calculate affinities between elements in $\mathbb{P}$. When a user $a \in A$ inserts the description of a resource d into an indexer $b \in B$, they send the pair $(M_A \bar{p}_n(d)^t, URL(d))$, where the first component is the projection of the resource profile and the second component is the URL of the resource. The details about this mechanism can be found in Sect. 3.1.

Next, an epidemic routing protocol occurs to distribute information inside the set of indexers from B. The objective of this epidemic protocol is to randomly spread the information in the resources through many different indexers. In this way, (i) the availability of the resource profiles increases, (ii) nodes in A may contact a random node $b \in B$ to perform queries without compromising the efficiency of the results; and (iii) the possible liability of providing access to a resource is shared among different nodes. There are many existing proposals of an epidemic protocol for locating resources in distributed systems [2, 5–7]. Indeed, nodes in B save the resource description of nodes in A, but since they have a user profile as well, it is possible to use this profile to organize nodes in B according to their interest, as in [7]. Thus, nodes in B can take advantage of the epidemic algorithms proposed in the literature, but they store and replicate the description of resources owned by nodes in A instead of the profiles of their own resources. These same epidemic routing mechanisms will be used to distribute query messages inside B.

2.1.3 QUERY and GET: Recommending Resources

Our security goal during this phase of the recommender system is to provide indexer anonymity, i.e., to make it impossible for an attacker to distinguish which indexer stored a particular answer to a query.

We describe the process in several steps. First, the customer issues a query to the indexers using an anonymous channel. Then, the indexers are automatically organized as a tree structure and calculate the parameter $\epsilon(\bar{d}, [\bar{q}])$ between the resources d they index and the query $\bar{q}$. As a result, the customer gets a vector of encrypted distances, decrypts these distances and chooses the indexes of resources that are more similar to the query. Next, the customer accesses the URLs of the selected resources using a PBR scheme.

QUERY resource profiles. A user $a \in A$ that searches for a resource in an indexer $b \in B$ builds a query $\bar{q}_n$ and chooses a private key K_a. This key is only known by a, which projects and encrypts the query to create $[\bar{q}]$. Next, a anonymously sends $[\bar{q}]$ to a random indexer $b \in B$, which calculates the parameter $e_j(\bar{p_j}, [\bar{q}])$ of every resource profile, and then creates a vector E_b that contains these parameters e_j. At the same time, the query is distributed to other indexers of the cluster B using an epidemic algorithm that creates a tree-shaped organization of random indexers. All the indexers in the tree answer the query, and all answers are joined together. Finally, a will receive an ordered set $E = \cup E_b$ where each component is the parameter $\epsilon(\bar{p}(r), [\bar{q}])$ of the resources that the nodes in B indexed. Next, a decrypts and calculates the distances to the resources, and selects those that are affine according to the threshold λ.

An epidemic algorithm without loops inside the cluster B, as we discussed earlier, can create the tree of random indexers. For the purposes of this resource, the tree structure has a disadvantage: answers provided by nodes in the inner branches of the tree will be located in the last positions of the joint vector E. Since we want to provide indexer anonymity, it is necessary for each node to locally permute the vector E. In this way, the customer cannot identify the position of an indexer in the tree according to the position of its answer within the vector E. The permutation that each indexer applies must be a secret that should not be made public.

GET URLs: Private Block Retrieval Protocol. Finally, a private block retrieval (PBR) scheme takes place in order to access the $URL(r)$ associated with the resource of their interest without leaking which specific $URL(r)$ the customer is obtaining.

The PBR hides the index of the item that the customer a is retrieving, and ensures that no one in the path $a \rightarrow b$ knows which item a is interested in, not even the indexer b. One of the PBR schemes suitable for our work is presented in [8]. The complexity of this algorithm is $O(k + j)$, being $k > \log(n)$ a security parameter and n the size of the database in bits. Some details about how indexers are organized to protect their identities will be introduced in Sect. 3.2.

2.1.4 ACCESS the Resource

Finally, the customer accesses the desired resource from the network. DocCloud includes a secure distributed file system that (i) makes not possible to learn any information about $\bar{p}(d)$ from $URL(d)$ of a resource, and (ii) it is not possible to access the $URL(d)$ from any node not in A without the group key K_A. Additional details about this file system can be found in [9], which describe secure distributed file systems appropriate for this use. In addition, Sect. 3.3 describes a mechanism to access resources using a streaming service.

3 Analysis of the Mechanisms in DocCloud

In this section, we analyze some of the mechanisms included in DocCloud to provide the services described in the previous section. Due to the lack of space, these analysis are going to be only an introduction to the work done in DocCloud. The interested reader will find more details in the references.

We include an analysis for these mechanisms: (i) distortion of the users' profile; (ii) protection of the intermediate nodes; and (iii) deploying a streaming service on the Internet of Things to access resources.

3.1 Projection of the Profiles

In DocCloud, merchants in a cluster of Things A insert the profiles of the resources they share into the indexers of a different cluster B. Since either these profiles include private information or private data may be inferred from them, profiles cannot be inserted directly into the system.

We define "profiles" $\bar{p}$ in a recommender system as an array of n real numbers. Each of the components of the profile captures the degree of interest of a user in a category using a real number from 0 to 1. The process of building these profiles is a complex task and falls beyond the scope of this chapter. If interested, the reader may refer to studies in the Information Retrieval and Artificial Intelligence fields [10].

We call $\mathbb{P}$ to the set of possible profiles. Our proposal is projecting profiles from $\mathbb{P}$ onto a new social space with fewer dimensions $m < n$ using a projection matrix $M_{m\times n}$. Thus, given $\bar{y}_m = M_{m\times n}\bar{p}_n$, an attacker is only able to calculate a class of original profiles $\hat{P}_n$ that solves the undetermined system:

$$\hat{P}_n = \bar{p}_p + c_1\bar{u}_1 + \ldots + c_{n-m}\bar{u}_{n-m}, \tag{1}$$

where $\bar{p}_p$ is any profile that verifies $\bar{y}_m = M_{m\times n} \cdot \bar{p}_p$, $c_1 \ldots c_{n-m}$ are arbitrary real numbers and $\bar{u}_1 \ldots \bar{u}_{n-m}$ are a basis of the kernel of $M_{m\times n} \cdot \bar{p} = 0$. The set of profiles that could be projected onto y_m is a subspace of dimension $n-m$, and attackers learn

that $\bar{p}_n \in \hat{P}$. If all profiles in the social space are equally likely, as the social model must enforce, then all profiles in $\hat{P}$ are equally likely and the attacker cannot learn any additional information from the projected profiles.

Two different problems arise in this scenario: (i) whether comparison of profiles makes sense in the projected space and (ii) the amount of information of the original profile that is preserved after the projection. We will draw on two lemmas. First, the Johnson-Lindestrauss' lemma [11]. According to this lemma, given a set of vectors of dimension n, it is possible to calculate a projection onto a metric space of dimension $m < n$ while limiting the error of the distances between projected vectors. Hence, two users that are affine in the original social space are also affine in the projected space with high probability. The second lemma we use for building our system is the undecomposability of random matrices [12]. According to this lemma, not only is the calculation of the exact original description not possible after projections, but also a malicious user will not be able to calculate a single component of the original profile if $n \geq 2m - 1$.

By means of these two lemmas, if we use a specially crafted matrix M to project a profile $p \in \mathbb{P}^n$ onto a profile $y = Mp^t \in \mathbb{P}^m$ where $n > 2m - 1$, then given y with high probability it is not possible to recover any component of p and the distances in the projected space are still related to the distances in the original space. We will test three projection matrices:

- A matrix with random components $m_{ij} \in_R [0, 1]$. We call this matrix M_R
- A matrix with components (the probabilities are discussed in Achlioptas [11]):

$$m_{ij} = \begin{cases} +1 & \textit{with probability } 1/6, \\ 0 & \textit{with probability } 2/3, \\ -1 & \textit{with probability } 1/6, \end{cases}$$

  We represent this matrix M_A. This matrix holds Johnson-Lindestrauss' lemma, as proved in [11].
- A hybrid matrix $M_H = pM_R + (1 - p)M_A$, where $p \in [0, 1]$

In the simulations that follow, we will use a social space of $n = 200$ categories. As a first approach, we will project onto space of $m = 20$ categories (the "20-space"). Given the projected profile and the projection matrix, a malicious user that tries to reconstruct the original profile has to solve a lineal system of 200 variables with 20 equations. There are 180 freedom degrees, and then we can safely establish that the original profile cannot be reconstructed. Under these circumstances, the privacy of the user is preserved, as we will show next.

Since the components of the vector (the interest of a user in a category) are real numbers between 0 and 1, the average profile is {0.5, 0.5, ..., 0.5}. The maximum distance from the average profile to any vector in the 200-space is, using the Euclidean metric, $d_{max} = \sqrt{200}/2 = 7.07$ and we will use this result to normalize the projected distances. We will use this maximum value for the distance in our simulations. Since we are interested in how the hybrid matrix behaves, we will use $p = 0.5$. The hybrid

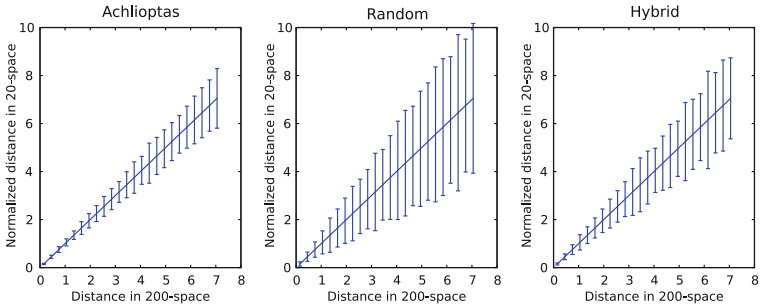


Fig. 2 Distances of the projected profiles using M_A, M_R and hybrid matrices

matrix approaches the behavior of M_A when $p \to 0$, and the random behavior when $p \to 1$.

Figure 2 shows the results of the projection of thousands of vectors using the three types of matrices Random, Achlioptas and Hybrid under study. The horizontal axis shows the distance between two vectors in the 200-space, while the vertical axis shows the average and standard deviation of the final distances between projected vectors into the 20-space. Figure 2 shows that there is a linear relation between distances. However, the standard deviation of the distance in the projected space increases when the distance in the original space increases.

Figure 2 shows that the standard deviation of the distances in 20-space increases with the distance in 200-space. This is very convenient, since it means that if two vectors are separated a long distance in the 200-space, then the region of possible distances in the 20-space is large. As a consequence, the estimation of the original distance in the 200-space given a distance in the 20-space is probabilistic, and user's privacy is preserved in a certain amount. We can use the standard deviation of the distance in m-space as a measure of the privacy achieved for each one of the matrices. Indeed, the larger this deviation, the larger is the region of distances that a given distance in the 200-space may project. We call this parameter the "uncertainty" of the distance, and it is a measure of the privacy of the proposal.

Proposition 1 *Given a vector in the n-space* $a \in \mathbb{P}^n$*, a distance* $d_n \in \mathbb{R}$*, a projection matrix into a m-space* M *and a set of vectors* $B = \{b \in \mathbb{P}^n | d(a, b) = d_n\}$*, the set of normalized distances* $D_m = \{d(a \cdot M, b \cdot M)\}$ *is a random variable where* $E[D_m] = d_n$*. We call uncertainty of the distance,* $U(n, d_n, m, M)$*:*

$$U(n, d_n, m, M) = 2 * \sqrt{E[(D_m - d_n)^2]} \tag{2}$$

Figure 3 shows the $U(n, d_n, m, M)$ of the different matrices for different values of m and M. The random matrix is nearly independent of the value of m, while the uncertainty of the Achlioptas matrix decreases when m increases. Furthermore, the Achlioptas matrix has much less uncertainty than the random matrix, as expected since it was created with this objective. While a high uncertainty is convenient to preserve privacy, it introduces a higher number of false positives and negatives. The

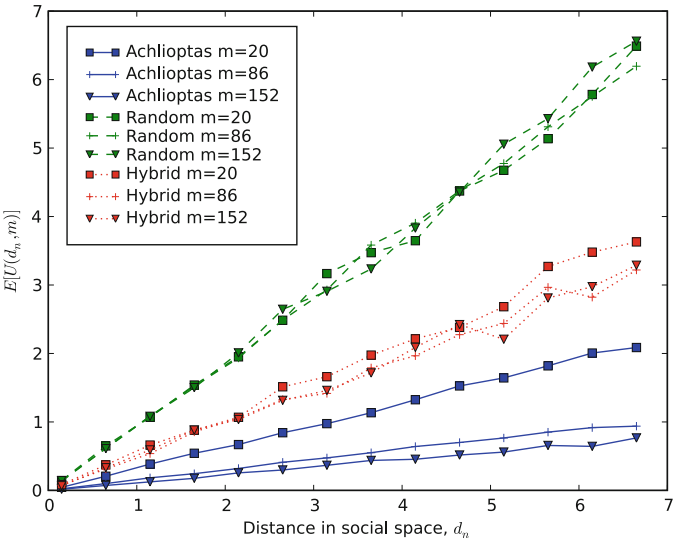


Fig. 3 Uncertainty of distances for different matrices

user can control this behavior by means of the parameter p of the hybrid matrix. In every matrix, the uncertainty increases linearly with d_n.

An interested reader can find additional information about these mechanisms in [13, 14].

3.2 A Metric for Indexer Anonymity

In this section, we will analyze the indexer anonymity property that the system shows, and we will provide a way to calculate the maximum number of items that each indexer must return to provide the plausible deniability to the indexers.

Indexers are organized in a tree as Fig. 4 shows. Indeed, indexers that are deeper in the tree structure send their profiles to their root, which performs a Bloom's filter to avoid repetition of profiles in the answer. The effect of this filter is that it is more likely that the information of a profile in the answer array comes from the leaves than from the root. In an extreme case, the root of the indexer's tree does not contribute at all to the answer. The attacker is not able to identify the indexer that holds a profile, but he can assign a different probability to each indexer in the tree. In this sense, the anonymity set is biased towards the inner leaves of the tree and it is smaller than expected.

We will consider an attacker that learns the list of resource's profiles that a given query returns. This may be the case of the sender of a query. We establish that the attacker wants to identify the indexer that stores a specific resource profile. To do

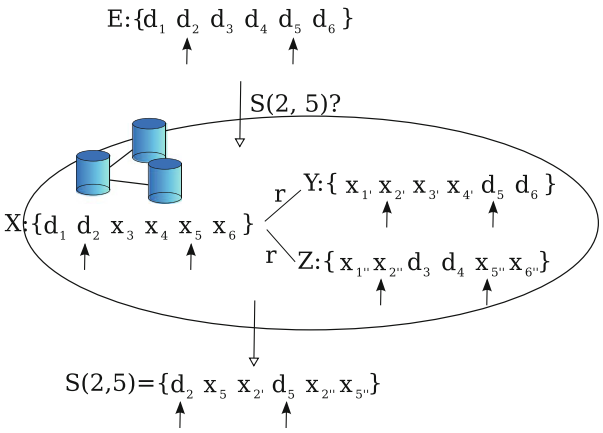


Fig. 4 A PBR scheme in the multi-indexer scenario

this, he issues a query that aims exactly to the targeted resource profile. We assume that the attacker knows the identity of the indexers that participated in the indexer tree. This is the case, for example, of a malicious indexer inside cluster B.

The main idea of this section is calculating the average size of the answer vector $a = |E|$ in the system of k indexers. Hence, we can force that if the indexer tree has k nodes, then each indexer contributes to E with a/k URLs. In this sense, from the point of view of the client, any resource profile could uniformly come from any of the indexers of the tree.

3.2.1 Uniform Assumption

Indexers of a cluster B store a set $D = \{d_1, d_2, ..., d_N\}$ of different resource profiles. Each indexer stores $n < N$ of them. When a query arrives, the recommender system will randomly pick a subset of indexers $S \subset B$ with cardinality k that contribute to the creation of the answer of the query, as explained in Sect. 2.1.

As a first approach, we suppose that resource's profiles are uniformly spread in B. That is to say, given a resource d_i, chances that an indexer stores d_i are independent of the indexer. For any indexer $b_u \in B$, we define an event $\bar{X}_{i,u}$ as "the resource d_i is not in b_u". As we assume uniform distribution of resources, the probability distribution function (pdf) of X can be modeled as a hypergeometric distribution: given a set of N different resources, $x = 1$ are of our interest. That is, d_i. Then, we pick without replacement n resources and calculate the chances that $k = 0$ of these are d_i.

$$p(\bar{X}_{i,u}) = hypergeom(x = 1; n = n, N = N, k = 0) \tag{3}$$

$$= \frac{\binom{1}{0}\binom{N-1}{n}}{\binom{N}{n}} = \frac{N-n}{N} \tag{4}$$

Given a subset S of k indexers, each one storing n different resource's profiles, we define the event $\bar{Y}_i$ as "the resource d_i is not in any of the indexers in S". The complement of this event, Y_i, means that the resource d_i is at least in one indexer in S. Since we assume a uniform distribution of resources, the pdf of Y_i is constant for any resource and indexer, and from this moment forward we will drop the subscript. Hence, the pdf of $\bar{Y}$ is:

$$pdf(\bar{Y}) = pdf(\bar{X})^k \tag{5}$$

$$pdf(Y) = 1 - pdf(\bar{Y}) = 1 - pdf(\bar{X})^k \tag{6}$$

Finally, we define an event Z_j as "the subset S has j different resources". Since each indexer stores n different resources, the minimum value of Z_j is n, that is to say, the k indexers are the same. The maximum value of Z_j is nk, and this is the case where the k indexers store completely different resources. Hence, the universe of Z_j is $[n, kn]$. This event is equivalent to "the subset S contents at least one instance of j resources and no instance of $N - j$".

Now, we can calculate the pdf of Z_j as follows.

$$pdf(Z_j) = \begin{cases} \binom{N}{j} pdf(\hat{Y})^{N-j}[1 - pdf(\hat{Y})]^j & \text{if } n \leq j \leq nk \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

$$= \begin{cases} \frac{N!}{N^{kN}} \frac{(N-n)^{k(N-j)} n^{ki}}{(N-j)!\, j!} & \text{if } n \leq j \leq nk \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The pdf of Eq. 8 is a binomial distribution that has been shifted by kn, and therefore its average is:

$$E[Z_j] = (k-1)n\ pdf(X) = \frac{(k-1)n(N-n)^k}{N^k} \tag{9}$$

Equation 9 captures the expected number of different items in the answered vector. The results of Eq. 9 can be used to improve the anonymity set of the indexers. Indeed, if each of the k indexers contributes with $n = E[Z_j]/k$ items, then the contribution of each indexer to the answer array is likely equal. In order to achieve this, we must encourage that $E[Z_j] = nk$. We call this n the optimal contribution coefficient for the indexers, n_{opt}, since it lets uniform contributions for the indexers and maximizes the anonymity of the set. We can calculate the optimal contribution of each indexer n_{opt} as the n that matches the following condition.

$$E[Z_j] = n_{opt}k = \frac{(k-1)n_{opt}(N - n_{opt})^k}{N^k} \tag{10}$$

$$k = \frac{(k-1)(N - n_{opt})^k}{N^k} \tag{11}$$

$$n_{opt} = N\left(1 - \sqrt[k]{\frac{k}{k-1}}\right) \tag{12}$$

Equation 12 shows the optimal contribution of each indexer to achieve maximum anonymity. Equation 11 shows the optimum number of indexers that must be contacted, for a fixed number of contributions from each indexer.

In the extreme case of $n, k \ll N$, $E[Z_j] \approx kn$ and in order to achieve uniform contributions, each indexer should contribute with $k \approx n$ items, nearly every item in the database. Since there are much more resources in the system than the capacity of an indexer, if the subset of indexers is small (k small), chances of collision are small and indexers can collaborate with every item.

Even if this could simplify the system design, it is not desirable from the point of view of efficiency. Users of the system will want to calculate the affinity to as many profiles as possible to be able to locate the more interesting resources. In this sense, the system will be designed for $kn \approx N$.

3.2.2 Epidemics Assumption

Actually, DocCloud includes an epidemic routing algorithm to distribute the database of available resources in the network during the procedure described in Sect. 2.1. The effect of this algorithm on resource's profiles is that it is much more likely for neighbor indexers to share similar resource profiles, and the likelihood of replicated data is higher if indexers are adjacent. Hence, in a real system the distribution of profiles is not uniform as we supposed in the last section, and the pdf that Eq. 3 shows will depend on the position of the indexers in the tree. Hence, $pdf(X_j)$ is not a simple hypergeometric distribution as calculated in the simplified scenario. On the contrary, $pdf(X_j)$ must be weighted with the position of the indexer.

As a first approach to analyze this problem, we are going to suppose that indexers are ordered in a line. This is a simplified tree with no branches. As in the last section, the event $\bar{X}_{j,u}$ represents "the resource r_j is not in d_u", but this time we define u as the position in line, from the root $u = 0$ to the branch $u = k$. Then, we describe a routing epidemic protocol in such a way that there is two real numbers ϵ and δ such as:

$$P(\bar{X}_{i,u}|\bar{X}_{i,u-1}) = p + \epsilon \tag{13}$$

$$P(\bar{X}_{i,u}|X_{i,u-1}) = p - \delta \tag{14}$$

$$0 \leq \epsilon + \delta \leq 1 \tag{15}$$

being p the probability of the uniform assumption that Eq. 3 shows. These equations may be interpreted as follows: the probability that a resource is (is not) in an indexer is higher if it is (is not) in the precedent indexer. Furthermore, we analyze a routing protocol that makes negligible the variation of likelihood of $X_{j,u}$ given $X_{j,v}$.

Equations 15 represents a Markov chain of probabilities. It was analyzed for example in [15], and we present next the solution for $P(X_{j,d})$ as a convenience using our notation.

$$P(\hat{X}_{i,u}) = \frac{(p-\delta) - (\epsilon+\delta)^u(\epsilon\delta - (1-p)\delta)}{1-\epsilon-\delta} \tag{16}$$

The last term of this equation attenuates with k, and then it is a monodic decreasing function with a maximum of p for $k = 1$. In the new scenario, $P(\bar{X}_{i,u}) \leq p$. The values for ϵ and δ cannot be easily computed since they depend on the actual epidemics algorithm in use, but we can conclude that any epidemics algorithm that we chose should use Eq. 12 as an upper limit for the contribution.

A real scenario with several branches in the indexer's tree is even more complex. The probability $P(X_{j,u})$ follows a Fisher's non-central hypergeometric distribution. In order to calculate the new pdfs or achieve similar conclusions to the last section, we need to model the epidemics algorithm that the indexer set uses. The specific k_{opt} that maximizes anonymity depends on the details of the epidemic algorithm that is used in the social network.

Even if we do not achieve a final result for the probability, we can extract some conclusions from the epidemics assumption. The Fisher's non-central hypergeometric distribution is always shifted toward the left and its average is less than the average of the central hypergeometric distribution. Besides, the analysis of this section for a simplified tree showed that the Markov chain that epidemic algorithm creates always decreases $P(X_{j,u})$. In practice, this means that we can use Eq. 12 as an upper limit for the amount of collaboration of the indexers of the system.

Figure 5 clarifies the analysis of this section. We used a recommender network that indexes $N = 1,000$ resources under the uniform assumption. The figure on the left represents the expected size of the answered vector for different sizes of the indexer tree. For a tree of $k = 10$ indexers, if each indexer contributes with $n = 80$ elements the answered vector has an expected length of 300 elements. The right side

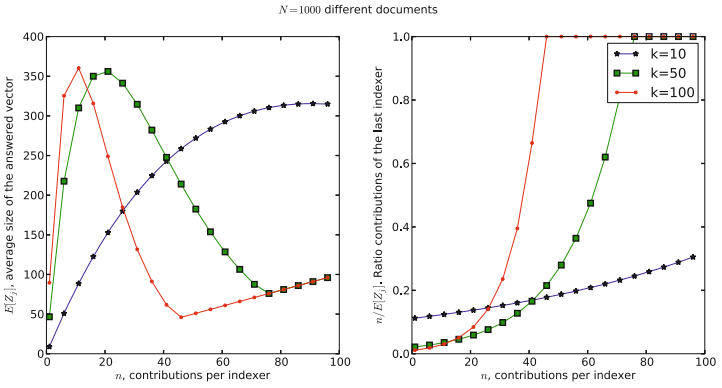


Fig. 5 Analysis of contributions for N=1,000 resources

of the figure shows the anonymity loss of the first indexer inside the tree. In the last example ($k = 10$, $n = 80$), the indexer in the inner leaves of the tree was the source of an item 0.2 of the time. Since the anonymity set of the tree has a size of $k = 10$ elements, attackers learn that the inner indexers are the source of an item about twice the time than in the maximum anonymity scenario.

For $k = 50$, the scenario is similar: the maximum length of the answered vector of affine resource profiles occurs when each node contributes with $n = 20$ items. In this case, the inner indexer is the source of an item with probability 0.05, when the maximum anonymity occurs at $1/k = 0.02$. In this case, the probability of an item to come from an indexer doubles the maximum anonymity scenario, and inner indexers in a tree of $k = 50$ indexers when each contribute with $n = 20$ items, are as protected as if the tree has only $k' = 25$ indexers.

These figures and the equations from this section can be used to decide the number of contributions from each indexer, the apparent size of the anonymity set and the expected size of the returned vector. Larger returned vectors enhance the efficiency of the system, since more affine resources are discovered during a query. But if the number of indexers in the indexer tree is not chosen accordingly, the anonymity loss of the indexers that first contribute to the answered vector may be unacceptably high.

An interested reader can find additional information about these mechanisms in [13, 14].

3.3 Streaming Services

One of the services offered by the different Things during the accessing phase is streaming. The protection of a streaming service is a non-trivial task. In this section, we explore how a scheme of oblivious databases for a private streaming service can be deployed on the Internet of Things. As a first approach, we explore the problem of accessing a single integer from the database, and next we generalize the problem to the selection of multimedia files.

3.3.1 Database as a Vector

A database stores N different integers less than 2^n for a known n, and the user wants to select the one with index j without leaking j to the database. To achieve this, the user prepares an array $\bar{s}(j)$ where each element is the Paillier encryption of 0, except the element j, that is the encryption of 1. That is to say,

$$\bar{s}(j) = \{s_0, s_1, \ldots, s_N\} \tag{17}$$

where

$$s_i = \begin{cases} [1] & \text{if } i = j \\ [0] & \text{if } i \neq j \end{cases} \tag{18}$$

We call this vector $\bar{s}(j)$ the **selection vector** of the element j. The user sends this selection vector to the database.

Next, the database multiplies each element s_i by the i-th element of the database, and adds all the resulting values. If the database is represented as a vector, $\bar{b} = \{b_0, \ldots, b_N\}$, it computes this operations:

$$S(j) = \sum s_i \otimes b_i \tag{19}$$

$$= [0] \otimes b_0 \oplus \ldots \oplus [1] \otimes b_j \oplus \ldots \oplus [0] \otimes b_N \tag{20}$$

$$= (\Pi(s_i^{b_i} \mod n^2)) \mod n \tag{21}$$

$$= [b_j] \tag{22}$$

And the database sends back the result $S(j)$ to the user, that decrypts this value to get b_j.

For the sake of clarity, we include next an example of this process. As a first approach, consider a vector without encryption $\bar{s}'(j) = \{0, 0 \ldots 1 \ldots 0, 0\}$. After computing the inner product, the database obtains a vector $\bar{S}(j) = \bar{s}'(j)\bar{b}^T = \{0, 0 \ldots b_j \ldots 0\}$, and after the addition of all elements, $S(j) = sum(\bar{S}) = b_j$. Equations (3–7) show these same operations with a encrypted selection vector, and then products and additions are on the encrypted text, as (5) shows. The result, finally, is the integer b_j that only the user that owns the private key of the Paillier's cryptosystem is able to decrypt.

3.3.2 Database as a Matrix

The mechanism that was described in the last section is absolutely inefficient if the user wishes to access a single, small element from the database. In fact, it is so inefficient that sending the whole database seems a better solution. The interested reader can find an analysis of this approach in [16]. Next, we adapt an enhancement of the proposal that was presented in [16] to our scenario.

In this case, the database organizes its elements using a square matrix of $\sqrt{N}$ rows and columns (we assume that $\sqrt{N}$ is integer) and the user follows the same protocol that was described in Sect. 3.3 to get not a single element, but a whole row of $\sqrt{N}$ elements. Hence, the database has $\sqrt{N}$ different rows $\bar{b} = \{r_1, ...r_{\sqrt{N}}\}$ where each row is a vector of $\sqrt{N}$ integers $r_i = (a_{i1}...a_{i\sqrt{N}})$. Now, the selection vector has $\sqrt{N}$ elements and it captures the row that the user is interested in. The system works in this way:

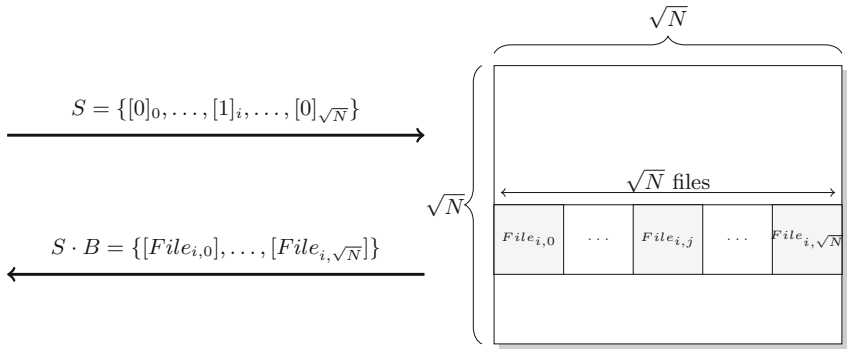


Fig. 6 Database as a matrix

$$S(j) = \sum s_i \otimes f_i \tag{23}$$

$$= (S(j, f_1), ..., S(j, f_{\sqrt{N}}) \tag{24}$$

$$= ([a_{j1}], ..., [a_{j\sqrt{N}}]) \tag{25}$$

This system is graphically represented in Fig. 6.

It could be tempting to take advantage of the additional elements that the user gets as a result of the scheme to avoid accessing new values if the previously accessed rows already included these values. In our opinion, the gain of efficiency in this situation does not pay off an unacceptable loss of privacy. As an example, let us imagine that the database orders audio files in such a way that every question related to the catholic religion is in the same row, while questions related to other religions are spread evenly on other rows of the database. If the user only performs one query to the database and makes use of the fact that every interesting question was included in the answer, then the database learns that the user is catholic. In this case, if the database knows that the user takes advantage of previous rows to safe some interactions, then it can be devised an ordering of questions suitable to learn the religion of the user. The only way to prevent this kind of attack is that the user knows the order of the interesting questions inside the database in advance to optimize the rows that it has to access. In a casual word, that seems hardly reasonable.

As a result, we conclude that in this case the user must prepare a query for every needed question and must not take advantage of previously accessed rows.

3.3.3 Generalization: From Integers to Files

We described a system that stores integers in a database. We wish to store in databases multimedia files, especially but not limited to audio files. If these files are cut down in packets and then each packet is represented as an unsigned integer, then the described schemes can be easily adapted to download whole files and not only single integers.

A simple division of files in packets is using bytes. In this case, a 64 KB file can be divided into 64,000 packets of integers from 0 to 255. Furthermore, all these packets that correspond to the same file share the same index inside the database. Hence, the user only needs to provide a single selection vector to privately download the 64,000 packets.

Many other PIR systems exist in the literature. Ostrovsky and Skeith [16] is a recent survey of many of them. Gentry and Ramzan [8] is the PIR system that as far as we know is more efficient for a single petition. Most of them are conceptually complex and difficult to code. We showed in [17] that the simple scheme works reasonably, the implementation is pretty fast and even performs better that the second, enhanced scheme in the scenario under study.

4 Conclusions

In this chapter, we introduced DocCloud, an example of a distributed service for the Internet of Things that protects the personal data exchanged in the network while provides personalized services.

First, DocCloud provides mechanisms to protect the profiles users share in the system for limiting the amount of private information these profiles show, but they are still are affine enough to receive useful recommendations. Also, DocCloud provides a private block retrieval scheme that connects customers and recommenders. This scheme ensures that recommenders cannot identify the profile of the resource they are providing to the user. This is not only a safeguard to protect the user's privacy; it also prevents recommenders from being prosecuted by aiding in the process of accessing a protected resource and provides intermediate nodes the security service of oblivious routing.

In addition, this chapter explores how the organization of databases in a tree-shaped structure prevents the identification of the source of the recommendation, and provides plausible deniability to databases. Not even the database knows whether or not it answered a specific query. We explored two different assumptions for the distribution of resource profiles inside the tree structure: a uniform distribution and a social distribution. The former is easier to analyze, but the latter is more similar to the organization of nodes in our recommender system. We provided an upper limit on the number of items that indexers must answer in order to provide optimal deniability inside the indexers tree.

There are some areas for improvement in this system. When a user gets the *URL* of an interesting resource, they still have to contact another network to actually access the resource. It is not clear whether or not the process of accessing can be separated from the process of selecting resources. For example, an attacker controls an indexer that forges special *URLs* in such a way that they are able to decide whether or not these are accessed afterward. In this way, they would be able to link a query to a user. A second open line of research involves the management of the social network. If cluster A is created only with users with similar profiles, then a "representative"

profile may be calculated for the cluster, and it may be close enough to the individual descriptions of each user to unacceptably leak private information that can be used to learn the users' profiles. In the complete description of DocCloud, we propose that clusters should be created with users with several "classes" of profiles. Users may show different profiles according to their current interests and join different clusters of the network at the same time. The impact of these "multi-ethnic" clusters on the efficiency of the system remains unclear. Additionally, although we were concerned about the protection of the user's privacy and introduced some mechanisms to provide this protection, we have not thoroughly analyzed the effect of these mechanisms on the efficiency of the recommendation process, and the amount of protection they provide.

This is a summary of the efforts inside the ARES project. Due to space constraints, most of the technologies introduced in this chapter are not detailed. The main results of this research were presented in [1, 9, 13, 14, 17].

References

1. Vera-del-Campo, J., Hernández-Serrano, J., Pegueroles, J., Soriano, M.: Design of a p2p content recommendation system using affinity networks. Comput. Commun. **36**(1), 90–104 (2012)
2. Euster, P., Guerraoui, R., Kermarrec, A.M., Maussoulie, L.: From epidemics to distributed computing. IEEE Comput. **37**(5), 60–67 (2004)
3. Schifanella, R., Panisson, A., Gena, C., Ruffo, G.: Mobhinter: epidemic collaborative filtering and self-organization in mobile ad-hoc networks. In: ACM Conference on Recommender Systems pp. 27–34. ACM, New York (2008)
4. Hernández-Serrano, J., Vera-del-Campo, J., Pegueroles, J., Gañán, C.: Low-cost group rekeying for unattended wireless sensor networks. Wireless Netw. **19**(2), 1–21 (2012)
5. Pouwelse, J., Yang, J., Meulpolder, M., Epema, D., Sips, H.: Buddycast: An operational peer-to-peer epidemoc protocol stack. In: 14th Annual Conference of the Advanced School for Computing and Imaging (2008)
6. Anglade, A., Tiemann, M., Vignoli, F.: Complex-network theoretic clustering for identifying groups of similar listeners in p2p systems. In: RecSys '07: Proceedings of the ACM Conference on Recommender Systems pp. 41–48. ACM, New York (2007)
7. Vera-del-Campo, J., Hernández-Serrano, J., Pegueroles, J.: Profile-based searches on p2p social networks. In: The Ninth International Conference on Networks, ICN. (2010)
8. Gentry, C., Ramzan, Z.: Single-database private information retrieval with constant communication rate. In: Automata, Languages and Programming. Lecture Notes in Computer Science, vol. 3580/2005, pp. 803–815. Springer, Berlin Heidelberg (2005)
9. Vera-del-Campo, J., Hernández-Serrano, J., Pegueroles, J.: Scfs: Design and implementation of a secure distributed filesystem. SECRYPT. (2008)
10. Manning, C.D., Raghadan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2009)
11. Achlioptas, D.: Database-friendly random projections: Johnson-lindenstrauss with binary coins. J. Comput. Syst. Sci. **66**(4), 671–687 (2003)
12. Liu, K., Kargupta, H., Ryan, J.: Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. IEEE Trans. Knowl. Data Eng. **18**(1), 92–106 (2006) (Senior Member-Kargupta, Hillol)
13. Vera-del-Campo, J., Hernández-Serrano, J., Pegueroles, J., Soriano, M.: Doccloud: a document recommender system on cloud computing with plausible deniability. Inform. Sci. **258**, 387–402 (2014)

14. del Campo, J.V.: In: Semantic overlay networks for P2P systems. Ph.D. Dissertation, Universitat Politecnica de Catalunya (2012)
15. Jaynes, E.T.: Probability Theory: The Logic of Science: Principles and Elementary Applications. Cambridge University Press, Cambridge (2003)
16. Ostrovsky, R., Skeith, I.W.E.: A survey of single-database private information retrieval: techniques and applications. In: Proceedings of the 10th International Conference on Practice and Theory in Public-Key Cryptography PKC'07, pp. 393–411. Springer-Verlag, Berlin, Heidelberg (2007)
17. Vera-del Campo, J., González-Muro, A., Soriano, M.: Private Audio Streaming for an Automated Phone Assistance System. In: Sixth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (2011)