
Barriers to the
implementation of
k-anonymity and
related microdata
anonymization techniques
in a realworld application

Barriers to the implementation of k-anonymity and related microdata anonymization techniques in a realworld application

Andreas Wiegand, 1878334
Ludwig Schallner, 1850413

1 Introduction

Nowadays data is a key factor in nearly every domain. It is comparable to the gold rush of the 19th. century [8]. Furthermore, storage space and network increasingly become affordable [10]. This is leading to the situation that the created and stored data is often not only useful to the original data holder, but to other researchers. Also, some data is only useful if its get shared with other data and get together analyzed. But those data may contains some personal or sensitive information. Such that the data should only get releases if the privacy is protected [6].

Table 1. Basic example

SSN	Age	Postcode	Problem
680-90-2665	25	4568	procrastination
008-07-4179	34	4567	stress
391-05-7998	48	4569	stomach cancer
078-36-3853	39	4568	obesity
411-71-9290	42	4561	stomach ulcers
527-59-1948	27	4568	stress

Data like table 1 have to get anonymized before release. A very common technique archive that goal, is the so called k-anonymity. Which goal is to prevent the possibility that information about the individual get leaked. This paper is showing the process of implementing k-anonymity into the real world. In Section 1 we will explain mandatory basic to understand k-anonymity and its purpose. Which leads to Section 2 the theoretical and heuristically implementation of k-anonymity. Section 3 will be discuss the underlying barriers of k-anonymity. In Section 4 we will explain to the reader multiply algorithms to implement k-anonymity and its barriers. A summary over the whole paper will be section 4

2 Basics

Microdata:

First of all, it should be clear what microdata is, those data is containing records of information about individuals. The upside versus the more known summary or aggregate data is, that microdata is naturally flexible. Everyone who has this data can perform own statistics from that data [1].

Identifier:

Attributes which can identify the record owner explicitly without any other attribute, for example full name (name and surname), telephone number, social security number. nicht sicher ob noch mehr möglich ist [4]

Quasi-identifier:

Even though explicit identifier got removed from published data. Attributes which non-explicitly identify the record owner are left. But if they get combined with other non-explicit attributes or other tables, they can reidentify the record owner. In such a case those combination of attributes are called quasi-identifier. For example Gender, Age, Postcode, weight and height [3]. Such process is shown in figure 1.

Sensitive data:

Data which is useful for example researchers but are private and should be known publicly nor be accessible for outsiders **Quelle**

Background knowledge:

Because its unknown what the attackers knows, we have to assume additionally to that he have access to table, the attackers knows that the table is generalized (to guarantee k-anonymity). Furthermore the attacks is aware of the domain of the attributes. **Instance-level background knowledge** The adversary knows that his target does know specific details about his target. For example Alice (the adversary) knows that Bob do not suffer from some disease, because he does not show the symptoms. In this case the adversary may can conclude what Bob is really suffers from.

Demographic background knowledge Adversary knows e.g $P(t[\text{condition}] = \text{cancer} \mid t[\text{Age}]_i \geq 40)$ May use it to interference about records **K-Anonymity** The goal of making a k-anonymized table, is to have at least (k-1) tuples of each identical tuple taking the corresponding quasi-identifiers into account [10, 6]. **equivalence class** Is a set of all tuples with the identical quasi-identifiers of a table [6].

3 theoretical and heuristically implantation of K-Anonymity

Another problem we will introduce is, that the producing of k-anonymity of a computational view is an NP-hard problem, like Meyerson and Williams shown. **NP Hard**

4 Underlying Barriers

A very basic underlying barrier of k-anonymity is, how to measure if a implan-tation has been successful or leads to a satisfying? A simple to calculate and to understand would be the calculation of distortion by Li, Wong, Fu and Pei [6].

Like Dalenius already mentioned it is absolutely necessary that an attacker, under no circumstances, can learn about whatsoever target if he is studying the published database. Not even if the attacker has background knowledge from any other sources [2]. Unfortunately like Dwork showed 2006 that such safety is impossible because of background knowledge. For example, if the attacker knows that Bob get paid twice as the average German man and the attacker got access to a database which publishes the average income by German men. The anonymity of Bob is compromised even if Bob's data is not in the database [5].

GRAMMAR CHECKED VIA grammarly.com ENDE VON NEUEN ZEUG

4.1 Linking data

A barrier to do the implementation of k-anonymity, the attacker can take another dataset and link both together to get rid off the k-anonymity and infer the real individual. This process is called linking data and was first described by Sweeney[10]. She showed that with a example of health care data from 37 states in the USA. The institute from which she bought the data, insures the anonymity of the individuals. Sweeney purchased the voter registration list for Cambridge Massachusetts and received information of the voters including ZIP code, birth date and gender (non explicit identifier) of each voter. She linked that information with the medical data. It was possible to deanonymize the data and get ethnicity, visit date, diagnosis, procedure, medication and total charge of some patients [10].

You got two datasets A and B. Each dataset got $\langle f_1, \dots, f_n \rangle$ features and $\langle r_1, \dots, r_n \rangle$ rows. Each row is then a tuple r_i with n features $\langle f_1, \dots, f_n \rangle$ describing the individual. Even tho the data is k-anonimized you can get rid oft he anonymity of the individual by linking the A to B. So if $A \cap B \neq \emptyset$ it is possible to infer the anonymized individual [10]. As a result any attacker who know such data (ZIP Code, Birth date and sex) could easily identify with such an attack his victim. For example Peter see his ex-wife at the doctor, most likely he knows her ZIP-Code, Birth date and sex. Therefore he finds out what she is suffer from.

4.2 Unsorted matching attack against k-anonymity

There is a possibility of a leak of information, if the release k-anonymity data is in some kind of a sort release. This mean the numerical attributes are descending or ascending sorted and attributes, which be of characters are alphabetical ordered, can give the attacker Information about the sensitive data. To prevent this attack, just get the data into a random order with a pseudo randomized sorting algorithm [10]. As an example take a look at the table 3: matching attack will

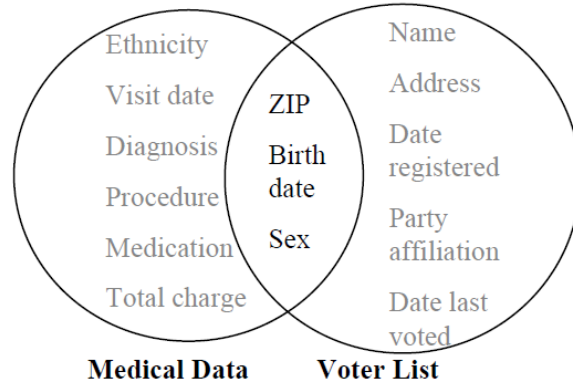


Fig. 1. linking data

give an example on that. If you compare the different release generalized tables you can figure out all quasi identifier of those [10].

...

Table 2. matching attack

Age	ZIP
2	91058
4	91058
50	27785
52	27785
20	32105
21	32105
31	67676
32	67676

Age	ZIP
*	91058
*	91058
5*	27785
5*	27785
2*	32105
2*	32105
3*	67676
3*	67676

Age	ZIP
2	91*
4	91*
50	27*
52	27*
20	32*
21	32*
31	67*
32	67*

4.3 Complexity of producing k-anonymity

Till now we only looked at problems of information leaking and privacy problems for individuals. Data is personal-specific information which is structured as a table in rows and columns. Rows a tuple. The columns are attributes with are a set of values which describe the certain attribute. A tuple specify a person. K-anonymity is about protecting the identity of a person not relationships of companies or governments. So the goal of k-anonymity is, not getting more information by linking the data to external data. The bridge between the data and external data is called "quasi-identifier". Examples for that would be ZIP, gender, birth date etc..

Generalization mean, replacing a value with a less specific but semantic identical value. For example we got a list of forenames of buys, (Achmed, Achilles, Achim). To generalize this names you can just (Ach*,Ach*, Ach*) delete the last chars of the name. So there is a less specific domain and now more generalize through this mapping. Suppression on the other hand means not releasing the value at all.

5 Algorithm

5.1 Clustering

Needed because data contains categorical values, the methods are not quite effective. [6] For calculating the weighted hierarchical distance of a cell which got generalized from level p to level q following formula is resulting to the solution.

$$WHD(p, q) = \frac{\sum_{j=q+1}^p \omega_{j,j-1}}{\sum_{j=2}^h \omega_{j,j-1}}$$

Distortions of generalization of tuples Distortions of generalization of tables

5.2 Datafly

5.3 Argus

proven that dont give k-anonymity in some cases by [9]!! where

References

1. Ipumsl-confidentiality, <https://web.archive.org/web/20070823010133/http://international.ipums.org/international/>
2. Dalenius, T.: Towards a methodology for statistical disclosure control. *Statistik Tidskrift* 15, 429–444 (1977)
3. Dalenius, T.: Finding a needle in a haystack or identifying anonymous census records. *Journal of official statistics* 2(3), 329 (1986)
4. Domingo-Ferrer, J., Torra, V.: A critique of k-anonymity and some of its enhancements. In: *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*. pp. 990–993. IEEE (2008)
5. Dwork, C.: Differential privacy. In: *Encyclopedia of Cryptography and Security*, pp. 338–340. Springer (2011)
6. Li, J., Wong, R.C.W., Fu, A.W.C., Pei, J.: Achieving k-anonymity by clustering in attribute hierarchical structures. In: *International Conference on Data Warehousing and Knowledge Discovery*. pp. 405–416. Springer (2006)
7. Machanavajjhala, A., Gehrke, J., Kifer, D., Venkitasubramaniam, M.: l-diversity: Privacy beyond k-anonymity. In: *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*. pp. 24–24. IEEE (2006)
8. Rossi, B.: Data revolution: the gold rush of the 21st century, <http://www.information-age.com/data-revolution-gold-rush-21st-century-2-123460039/>
9. Sweeney, L.: Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10(05), 571–588 (2002)
10. Sweeney, L.: k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10(05), 557–570 (2002)