University Passau
Faculty of Computer Science and Mathematics

Bachelor Thesis

Privacy Model Substitution

Thesis Prepared for the Degree of
Bachelor of Science (B.Sc.)

at the Chair of Distributed Information Systems
of the Faculty of Computer Science and Mathematics
of the University Passau

Name:	Fabian Pfeil
Matriculation Number:	77560
Subject Area:	Computer Science
Course of studies:	Bachelor Internet Computing
Schwerpunkt:	TODO
Studienjahrgang:	TODO
Erstprüfer:	Prof. Dr. NN
Zweitprüfer:	Prof. Dr. NN2

Contents

List of Figures

List of Tables

Abstract

Zitiertest[11].
K-anonymity[17].
L-diversity[14].
T-closeness[12].
d-presence[15]
delta-disclosure[4]
basic-beta-likeness[5]
k-map1[9]
population uniqueness[6]
profitability 1[18]
profitability 2[16]
arx[1]
diff[8]
diff2[3]
diff3[13]

1 Introduction

1.1 Motivation

Figure 1.1: Describe this picture.

1.2 Layered Privacy Language

2 Privacy Model Classification

In this section, we will take a look at the Privacy Models implemented by Arx. Therefore, these Privacy Models will be explained in terms of requirements for the data sets to which they can be applied, the attacks they mitigate, their use cases, advantages and disadvantages.

2.1 Definitions

2.2 K-Anonymity

The first Privacy Model, we will take a look at is K-Anonymity. This Privacy Model was released in 2002 by Latanya Sweeney with the goal to prevent re-identification of data subjects of certain data sets[17]. K-Anonymity is defined as follows:

> "Let RT(A_1,...,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}]."[17]

To clarify what this means, figure 2.1 shows the example used in Sweeney's work[17]. The figure shown here, is an example for a table that satisfies the k-anonymity criterion. The quasi-identifier for this particular case is QI_T= {Race, Birth, Gender, ZIP} and k = 2. This means that every tuple of quasi-identifying attributes appears at least in two records in T.
As k-Anonymity is defined, we will now take a look at the attacks, which can be mitigated with the property of k-anonymity applied. As this issue was already addressed by Fung et al., their classification will be used here. In the work of Fung et al. it is stated that k-anonymity, applied to a data set, will prevent only Record Linkage attacks. Consequently, other attacks like Attribute Linkage, Table Linkage or a Probabilistic Attack can not be mitigated by k-Anonymity[10].

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

Figure 2.1: Example of a 2-anonymous table [17]

2.2.1 K-Map

The Privacy Model of K-Map is directly related to k-Anonymity. El Emam et al. show that extending k-Anonymity to k-Map can however reduce the loss of information due to over-anonymization of the dataset[9]. Damien Desfontaines describes k-Map like this: "Your data satisfies k-map if every combination of values for the quasi-identifiers appears at least k times in the reidentification dataset."[7]
The similarity to k-Anonymity is obvious, as the main criterion in both Privacy Models is the same, the only difference is the data set on which they are based. In the case of k-Map it is not the data set of the data custodian, it is the reidentification dataset[7]. Consequently, K-Map mitigates the same attacks (Record Linkage Attacks) as k-Anonymity. Other attacks can not be prevented by this Privacy Model.
However, the work of El Emam et al. states, that even though k-Map reduces information loss in comparison to k-Anonymity, the model of k-Map is not used in practices because one can assume that a data custodian does not have access to a reidentification dataset, while an attacker does[9].

2.3 l-Diversity

The next Privacy Model we will take a look at is l-Diversity. L-Diversity was proposed by Machanavajjhala et al. to overcome the weaknesses of k-Anonymity, namely the attribute linkage attacks[14]. To understand why l-Diversity mitigates attribute linkage, we will take a short look at the definition. In the work of Machanavajjhala et al. the principle of l-Diverity is defined as follows:

> "A q^*-block is l-diverse if it contains at least l well-represented values for the sensitive attribute S. A table is l-diverse if every q^*-block is l-diverse." [14]

This means that every group of quasi-identifiers has to at least contain l different values for the sensitive attribute S. It is because of this property that l-diversity can prevent attribute linkage attacks. However, table linkage attacks can not be mitigated through l-diversity[10]. Machanavajjhala et al. use the table shown in figure 2.2 as an example for l-diversity. In this example, one can see that every group of quasi-identifiers contains at least three different values for the sensitive attribute.

	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

Figure 2.2: Example of a 3-diverse table [14]

There are three instantiations of l-diversity implemented in Arx. Consequently, we will only take a look at these.

2.3.1 Distinct-l-Diversity

Distinct-l-Diversity is arguably the simplest of the three instantiations of l-diversity. It is also known as p-sensitive k-anonymity. Distinct-l-diversity uses the plain definition of l-diversity mentioned above. Therefore, no advanced calculation of the l-factor has to take place. One can simply take the number of different values of the sensitive attribute as the factor l[10].
Lets get back to our example in figure 2.2. With distinct-l-diversity the table shown here is, as simple as it sounds, 3-diverse because every group of quasi-identifiers has three different values. Consequently, this table is diverse.
With this form of l-diversity, probabilistic inference attacks cannot be mitigated, because of the fact that some values of sensitive attributes are more

frequent than others. Therefore, two stronger instantiations of l-diversity have been created[10].

2.3.2 Entropy-l-Diversity

The next instantiation of l-Diversity is Entropy-l-Diversity. It is defined as follows:

> "A table is Entropy l-Diverse if, for every q^*-block,
>
> $$-\sum_{s \in S} p(q^*, s')log(p(q^*, s')) \geq log(l)$$
>
> where
>
> $$p(q^*, s) = \frac{n(q^*, s)}{\sum_{s' \in S} n(q^*, s')}$$
>
> is the fraction of tuples in the q^*-block with sensitive attribute value equal to s."[14]

With this calculation of the l-factor for our example in figure 2.2, our table is actually 2.8-diverse. Consequently, every group of quasi-identifiers has at least 2.8 different values for the sensitive attribute. And as there can not be 2.8 different values there are in our case three existing values.

2.3.3 Recursive (c, l)-Diversity

The third and final instantiation of l-Diversity we will take a look at is the recursive (c, l)-Diversity. This privacy model makes sure, that the most frequent values of sensitive attributes do not appear to often in the table. Furthermore it makes the uncommon values appear not too rarely[10]. Machanavajjhala et al. define recursive (c, l)-Diversity like this:

> "In a given q^*-block, let r_i denote the number of times the i^{th} most-frequent sensitive value appears in that q^*-block. Given a constant c, the q^*-block satisfies recursive (c, l)-diversity if r1 $>$ c(r_l + r_{l+1} + ... + r_m). A table T^* satisfies recursive (c, l)-diversity, if every q^*-block satisfies recursive l-diversity. We say that 1-diversity is always satisfied."[14]

Both of these definitions for l-diversity, entropy and recursive, may be too restrictive. For entropy-l-diversity this can bee seen in the fact that, if entropy l-diversity should be applied, the entropy of whole table has to be log(l) or higher. This may be very difficult to achieve when, for example, a value for a

sensitive attribute is too common.
The same is the case for recursive (c, l)-Diversity. When for example a sensitive attribute value is present 90% of the time and the factor c is chosen < 9, it is impossible to achieve recursive (c, l)-Diversity[14].
This is not the only Problem with l-Diversity. Sensitive information can be leaked, despite the fact that l-diversity is applied, because l-diversity only guarantees the diversity of sensitive attribute values, but does not take the semantical relations of these values into account. Consequently, l-Diversity is insufficient to prevent attribute linkage in some cases where the distribution of a sensitive attribute is skewed[12].

2.4 t-Closeness

The next privacy model is t-Closeness. It was proposed by Li et al. to prevent attribute linkage through skewness attacks, mentioned earlier in section 2.3.3 [12]. In the work of Li et al. the property of t-Closeness is defined as follows:

> "An equivalence class is said to have t-closeness if the distance between the distribution of a sensitive attribute in this class and the distribution of the attribute in the whole table is no more than a threshold t. A table is said to have t-closeness if all equivalence classes have t-closeness."[12]

The factor t can be used to manage a trade off between privacy and utility. To measure the distance between those distributions and to compute the t-factor, Li et al. use the Earth Mover's Distance (EMD). In the work of Li et al, certain formulas were derived to calculate the EMD for the cases we will consider in the following subsections of t-closeness[12].

2.4.1 Ordered Distance t-Closeness

This type of t-closeness is used for numerical attributes, as these can be ordered. The distance between the distributions (in this case P and Q, with $r_i = p_i - q_i, (i = 1, 2, ...m)$) is calculated as follows [12]:

$$D[P,Q] = \frac{1}{m-1}(|r_1| + |r_1 + r_2| + ... + |r_1 + r_2 + ...r_{m-1}|) = \frac{1}{m-1}\sum_{i=1}^{i=m}\left|\sum_{j=1}^{j=i} r_j\right|$$

2.4.2 Equal Distance t-Closeness

Some types of attributes can not be orered like numerical values. Therefore, the next two computations of the EMD are used for categorical attributes, as they can not be ordered.
With the equal distance approach, all distances between any two values of categorical attributes are considered to be 1. Consequently, the following formula to compute the distance is the result[12]:

$$D[P,Q] = \frac{1}{2}\sum_{i=1}^{m}|p_i - q_i| = \sum_{p_i \geq q_i}(p_i - q_i) = -\sum_{p_i < q_i}(p_i - q_i)$$

2.4.3 Hierarchical Distance t-Closeness

Another way to compute the distance between distributions of categorical values is the hierarchical distance approach. This technique bases the distances of values on the minimum level to which these two values can be generalized to, based on a domain hierarchy[12].
Therefore, the so called extra of a leaf of this hierarchy is defined as:

$$extra(N) = \begin{cases} p_i - q_i & \text{if N is a leaf} \\ \sum_{C \in Child(N)} extra(C) & \text{otherwise} \end{cases}$$

where Child(N) is the portion of leaf nodes below N. Furthermore, two additional functions for internal Nodes are defined:

$$pos\_extra(N) = \sum_{C \in Child(N) \wedge extra(C) > 0} |extra(C)|$$

$$neg\_extra(N) = \sum_{C \in Child(N) \wedge extra(C) < 0} |extra(C)|$$

This leads to the cost-function, which describes the cost of moving between some leaf's (N) children branches.

$$cost(N) = \frac{height(N)}{H} min(pos\_extra(N), neg\_extra(N))$$

Finally, the EMD can be seen as follows:

$$D[P,Q] = \sum_{N} cost(N)$$

Now that we have defined the special cases for t-closeness, we will take a look at a short example used by Li et al. in their work[12].

	ZIP Code	Age	Salary	Disease
1	4767*	≤ 40	3K	gastric ulcer
3	4767*	≤ 40	5K	stomach cancer
8	4767*	≤ 40	9K	pneumonia
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
2	4760*	≤ 40	4K	gastritis
7	4760*	≤ 40	7K	bronchitis
9	4760*	≤ 40	10K	stomach cancer

Figure 2.3: Table that has 0.167-closeness w.r.t. Salary and 0.278-closeness w.r.t. Disease [12]

Figure 2.3, shows a Table which has 0.167-closeness in relation to the sensitive attribute Salary, which can be calculated using the ordered distance formula for numerical values, and 0.278-closeness in relation to the attribute Disease. This can be easily computed using the given domain hierarchy shown in figure 2.4.

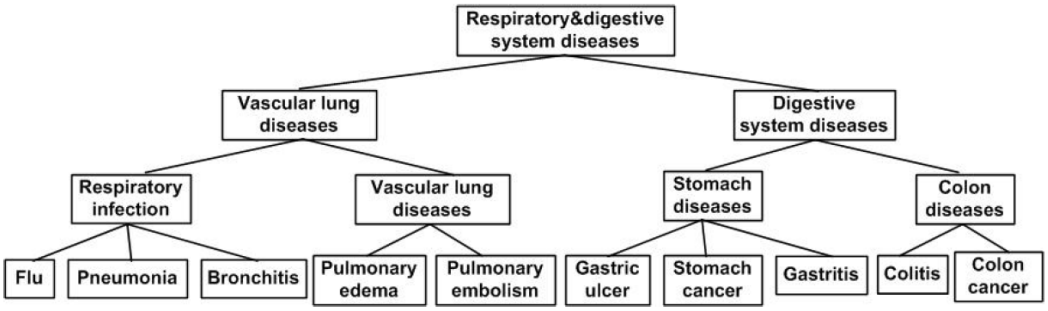

Figure 2.4: Hierarchy for categorical attributes Disease [12]

2.5 delta-Disclosure privacy

The next privacy model is delta-disclosure privacy. It also can be used to protect data sets from attribute linkage attacks, due to the fact that it is related to the t-closeness model. It also enforces restrictions on the distances between the sensitive attribute distributions. In contrast to the model proposed by Li et al., delta-disclosure privacy uses a multiplicative approach, which makes it stricter than t-closeness[4].
Delta-disclosure privacy was proposed by Brickell et al. and is defined as follows:

"We say that an equivalence class $\langle t \rangle$ is δ-disclosure-private with regard to the sensitive attribute S if, for all s $\in$ S

$$A_{quot}(\langle t \rangle) = \left| log \frac{p(\langle t \rangle, s)}{p(T, s)} \right| < \delta$$

A table T is δ-disclosure-private if for every t $\in E_Q$, $\langle t \rangle$ is δ-disclosure private."[4]

In easier words, one can say that a table T is delta-disclosure private if the distributions of sensitive attributes in the equivalence classes and in the overall table are approximately the same[4].

2.6 beta-Likeness

The privacy model beta-Likeness tries to overcome the limitations of the previous two models, as the EMD in t-closeness does not provide a clear privacy guarantee and the delta-disclosure privacy requires that every value of a sensitive attribute of the overall table occurs in every equivalence class. This makes delta-disclosure privacy unnecessarily strict.
To avoid these disadvantages, Cao et al. propose a privacy model called beta-Likeness. This model assumes that the distribution of the sensitive attributes in a table are known to the public and bases it's privacy constraint on information gain, which is denoted as a difference function between the distribution of a sensitive attribute in the overall table and in an equivalence class[5].
In the work of Cao et al., two instantiations of beta-Likeness are introduced.

2.6.1 basic beta-Likeness

The base approach for beta-likeness is defined as follows:

"Given table DB with sensitive attribute SA, let V = $\{v_1$, ... , $v_m\}$ be the SA domain, and P = (p_1, ... , p_m) the overall SA distribution in DB. An EC G with SA distribution Q = (q_1, . . . , q_m) is said to satisfy basic β-likeness, if and only if $max\{D(p_i, q_i) | p_i \in P, p_i < q_i\} \leq \beta$, where $\beta > 0$ is a threshold."[5]

This means, that every distance between the distributions of sensitive attributes in the overall table and in the equivalence class has to be lower or equal to a certain threshold β.

Furthermore, Cao et al. state that every equivalence class of an anonymized table satisfies beta-likeness, the whole table obeys beta-likeness.
The distance function for this privacy model is defined as $D(p_i, q_i) = \frac{q_i - p_i}{p_i}$ as Cao et al. opt for a relative difference instead of a absolute difference, because it does not suite their purposes. This relative distance function pays attention to less frequent values of sensitive attributes. Consequently, sensitive attribute values with a large frequency were not put into consideration. To mitigate the privacy threat caused by this, Cao et al. provide a stronger definition of beta-Likeness[5].

2.6.2 Enhanced beta-Likeness

This instantiation of beta-Likeness is stronger than basic beta-Likeness, as the name suggests. Enhanced beta-Likeness is defined as follows:

> "For table DB with sensitive attribute SA, let V = $\{v_1, ..., v_m\}$ be the SA domain, and P = $(p_1, ..., p_m)$ the overall SA distribution in DB. An EC G with SA distribution Q = $(q_1, ..., q_m)$ is said to satisfy enhanced β-likeness, if and only if $\forall q_i, D(p_i, q_i) = \frac{q_i - p_i}{p_i} \leq min\{\beta, -\ln p_i\}$, where $\beta > 0$ is a threshold and $\ln p_i$ is the natural logarithm of p_i."[5]

The properties that come with this definition protect the privacy for al sensitive attribute values. Values with rare occurrence receive sufficient attention, while values that occur more often can not approach frequency values of 1. As it is more robust than basic beta-likeness in terms of privacy, it should be used instead of it's predecessor[5].
Due to these traits and the fact that it is related to t-closeness and delta-disclosure privacy, the privacy model of beta-likeness can protect data sets against attribute linkage attacks.

2.7 delta-Presence

The next privacy model that we take into consideration is delta-Presence. This model was proposed by Nergiz et al. to prevent attacks from identifying that a certain suspect is part of a dataset (Table Linkage), as this can pose a serious privacy threat in certain cases[15].

The privacy model of delta-presence is defined as follows:

> "Given an external public table P, and a private table T , we say that δ-presence holds for a generalization T* of T, with $\delta = (\delta_{min}, \delta_{max})$ if
>
> $$\delta_{min} \leq P(t \in T|T*) \leq \delta_{max} \quad \forall t \in P"[15]$$

In a dataset which applies this privacy criterion, every tuple t is called δ-present within the range of $\delta = (\delta_{min}, \delta_{max})$. To clarify what this means, we will take a short look at the example used in the work of Nergiz et al.
In Figure 2.5 the tables $P*_3$ and $T*_3$ are given. To get the probabilities $\delta_{min}, \delta_{max}$, the following calculations are done. $P(a \in T|T*_3) = \frac{|b,c,f|}{|a,b,c,d,e,f|} = \frac{1}{2}$. The same is done for tuples b,c,d,e and f. The probability for the tuples g,h and i is calculated with $\frac{|h,i|}{|g,h,i|} = \frac{2}{3}$. Consequently, the probability that a tuple from Table $P*_3$ is also in $T*_3$ lies between $\frac{1}{2}$ and $\frac{2}{3}$.
Like in most privacy models the δ-factor can be used as a trade-off between privacy and utility. Therefore this factor has to be chosen carefully.

P_3^*

	Public Dataset			Sen.
	Zip	Age	Nationality	
a	47*	*	America	0
b	47*	*	America	1
c	47*	*	America	1
d	47*	*	America	0
e	47*	*	America	0
f	47*	*	America	1
g	48*	*	Europe	0
h	48*	*	Europe	1
i	48*	*	Europe	1

T_3^*

	Research Subset		
	Zip	Age	Nationality
b	47*	*	America
c	47*	*	America
f	47*	*	America
h	48*	*	Europe
i	48*	*	Europe

Figure 2.5: Table $T*_3$ that is $(\frac{1}{2}, \frac{2}{3})$-present [15]

2.7.1 Inclusion

Only mentioned in Github arx, allows to define a subset.

2.8 Profitability

Now we will take a look at a group of privacy models that are part of the so called game theoretic approach, which was proposed by Wan et al in 2015. This approach tries to find the best possible de-identification strategy to maximize the data publishers monetary gain[18].
In the implementation of arx, four privacy models are present which represent

two scenarios. In the *Prosecutor* model, we assume that the attacker knows that a certain record is present in our data base. Consequently groups have to be formed. This is realized by the k-anonymity privacy model, mentioned in section 2.2.
In the second scenario, the *Journalist* model, the assumption that a attacker knows about an individual's membership is not made. Therefore groups can be made by generalizing the population table. This corresponds to the k-map privacy model mentioned in section 2.2.1[16].
For every scenario mentioned here, two variants are implemented by arx. The first one is the SH-Friendly variant which uses generalization hierarchies and appropriate levels of minimal generalization to always satisfy the Safe Harbor policy of HIPAA. The second one is the No-Attack variant. This model guarantees that the adversary never has any urge to even try to attack the published data. Therefore an attacker will only attack, if his monetary gain for attacking a record is greater than the costs he has to encounter. This can be described as:

$$SP(r) \cdot gain > cost$$

Consequently the cost of the attack always has to be greater than the attackers estimated monetary gain.[16]
It is because of the use of k-anonymity and k-map, that the privacy models of Profitability can be used to protect data from Record Linkage while also maximizing the users monetary gain.
The specifications above result in four privacy models:

- ProfitabilityProsecutor
- ProfitabilityJournalist
- ProfitabilityProsecutor No Attack
- ProfitabilityJournalist No Attack

2.9 Differential privacy

The next privacy model is the model of differential privacy, first proposed by Cynthia Dwork in 2006, does not focus on the output dataset on its own but on the data processing method [8, 2]. This model ensures that a disclosure by any attacker is just as likely whether or not one individual does participate in a data base[8].
The privacy model implemented in arx is the model of (ϵ, δ)-Differential Privacy, which is an extension of ϵ-Differential Privacy. ϵ-Differential Privacy can

be defined as follows:

> "A randomized function K gives ϵ-differential privacy if for all data sets D_1 and D_2 differing on at most one element, and all S $\subseteq$ Range(K),
>
> $$Pr\left[K(D_1) \in S\right] \leq exp(\epsilon) \times Pr\left[K(D_2) \in S\right]\text{"}[8]$$

As this definition of ϵ-Differential Privacy can be too strict in practice, a commonly used relaxation was introduced, namely (ϵ, δ)-Differential Privacy. This version of differential privacy allows a small error probability δ and is defined as follows[13]:

> "A randomized algorithm A satisfies (ϵ, δ)-differential privacy, if for any pair of neighboring datasets D and D' and for any O $\subseteq$ Range(A):
>
> $$Pr\left[A(D) \in O\right] \leq e^{\epsilon} \times Pr\left[A(D') \in O\right] + \delta\text{"}[13]$$

Due to the fact that (ϵ, δ)-Differential Privacy allows a small error probability, a higher data quality is possible in comparison to the traditional ϵ-Differential Privacy[3]. With this privacy model applied it becomes very difficult for adversaries to gain any information about specific individuals. Consequently, data sets are protected from Record Linkage, Attribute Linkage and Table Linkage[2].

2.10 Average risk

2.11 Population uniqueness

2.12 Sample uniqueness

2.13 Classification Table

The results from chapter 2 are shown in the following classification table. The attacks mitigated by each privacy model are displayed here in table 2.1.

Privacy Models	Attacks Model			
	Record Linkage	Attribute Linkage	Table Linkage	Probabilistic Attack
k-Anonymity	X			
k-Map	X			
Distinct-l-Diversity	X	X		
Entropy-l-Diversity	X	X		
Recursive (c, l)-Diversity	X	X		
Ordered Distance t-Closeness		X		X
Equal Distance t-Closeness		X		X
Hierarchical Distance t-Closeness		X		X
delta-Disclosure Privacy		X		X
Basic beta-Likeness		X		X
Enhanced beta-Likeness		X		X
delta-Presence			X	
Inclusion				
Profitability Prosecutor	X			
Profitability Journalist	X			
Profitability Prosecutor No Attack	X			
Profitability Journalist No Attack	X			
(e, d)-differential Privacy	X	X	X	
Average Risk	X			
Population Uniqueness	X			
Sample Uniqueness	X			

Table 2.1: Attacks mitigated by each privacy model

3 Evaluation of Privacy Models (Performance)

4 Privacy Model Substitution Table

5 Conclusion

6 Bibliography

[1] arx-deidentifier/arx. https://github.com/arx-deidentifier/arx/tree/master/src/main/org/deidentifier/arx/criteria. Last accessed 15 October 2018.

[2] Privacy models. https://arx.deidentifier.org/overview/privacy-criteria/. Last accessed 15 October 2018.

[3] Raffael Bild, Klaus A. Kuhn, and Fabian Prasser. SafePub: A truthful data anonymization algorithm with strong privacy guarantees. *Proceedings on Privacy Enhancing Technologies*, 2018(1):67–87, jan 2018.

[4] Justin Brickell and Vitaly Shmatikov. The cost of privacy. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 08*. ACM Press, 2008.

[5] Jianneng Cao and Panagiotis Karras. Publishing microdata with a robust privacy guarantee. *Proceedings of the VLDB Endowment*, 5(11):1388–1399, jul 2012.

[6] Fida Kamal Dankar, Khaled El Emam, Angelica Neisa, and Tyson Roffey. Estimating the re-identification risk of clinical data sets. *BMC Medical Informatics and Decision Making*, 12(1), jul 2012.

[7] Damien Desfontaines. k-map, the weird cousin of k-anonymity - ted is writing things. https://desfontain.es/privacy/k-map.html. Last accessed 15 October 2018.

[8] Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming*, pages 1–12. Springer Berlin Heidelberg, 2006.

[9] Khaled El Emam and Fida Kamal Dankar. Protecting privacy using k-anonymity. *Journal of the American Medical Informatics Association*, 15(5):627–637, sep 2008.

[10] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving data publishing. *ACM Computing Surveys*, 42(4):1–53, jun 2010.

[11] Armin Gerl, Nadia Bennani, Harald Kosch, and Lionel Brunie. LPL, towards a GDPR-compliant privacy language: Formal definition and usage. In *Lecture Notes in Computer Science*, pages 41–80. Springer Berlin Heidelberg, 2018.

[12] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *2007 IEEE 23rd International Conference on Data Engineering*. IEEE, apr 2007.

[13] Ninghui Li, Wahbeh Qardaji, and Dong Su. On sampling, anonymization, and differential privacy or,k-anonymization meets differential privacy. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security - ASIACCS '12*. ACM Press, 2012.

[14] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. L-diversity: privacy beyond k-anonymity. In *22nd International Conference on Data Engineering (ICDE'06)*. IEEE, 2006.

[15] Mehmet Ercan Nergiz, Maurizio Atzori, and Chris Clifton. Hiding the presence of individuals from shared databases. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data - SIGMOD '07*. ACM Press, 2007.

[16] Fabian Prasser, James Gaupp, Zhiyu Wan, Weiyi Xia, Yevgeniy Vorobeychik, Murat Kantarcioglu, Klaus Kuhn, and Brad Malin. An open source tool for game theoretic health data de-identification. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2017:1430–1439, 2017.

[17] Latanya Sweeney. k-anonymity: a model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, oct 2002.

[18] Zhiyu Wan, Yevgeniy Vorobeychik, Weiyi Xia, Ellen Wright Clayton, Murat Kantarcioglu, Ranjit Ganta, Raymond Heatherly, and Bradley A. Malin. A game theoretic framework for analyzing re-identification risk. 10:e0120592.

Erklärung zur Bachelorarbeit

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Passau, den <date>

<First Name, Last Name>