On the Evaluation of the Privacy Breach in Disassociated Set-valued Datasets

Sara Barakat[1], Bechara Al Bouna[1], Mohamed Nassar[2] and Christophe Guyeux[3]

[1]*TICKET Lab., Antonine University, Hadat-Baabda, Lebanon*

[2]*Department of Computer Science, MUBS, Hamra, Lebanon*

[3]*University of Bourgogne Franche-Comté, Belfort, France*

Keywords: Disassociation, Privacy Breach, Data Privacy, Set-valued, Privacy Preserving.

Abstract: Data anonymization is gaining much attention these days as it provides the fundamental requirements to safely outsource datasets containing identifying information. While some techniques add noise to protect privacy others use generalization to hide the link between sensitive and non-sensitive information or separate the dataset into clusters to gain more utility. In the latter, often referred to as bucketization, data values are kept intact, only the link is hidden to maximize the utility. In this paper, we showcase the limits of disassociation, a bucketization technique that divides a set-valued dataset into k^m-anonymous clusters. We demonstrate that a privacy breach might occur if the disassociated dataset is subject to a cover problem. We finally evaluate the privacy breach using the quantitative privacy breach detection algorithm on real disassociated datasets.

1 INTRODUCTION

Privacy preservation is an important requirement that must be considered carefully before the release of datasets containing valuable information. Anonymizing a dataset by simply removing the personally identifying information (PII) is insufficient to prevent a privacy breach (Samarati, 2001; Sweeney, 2002). An attacker[1] may still succeed in associating his/her background knowledge with one or multiple records via non-sensitive information in the dataset to eventually reveal individuals' sensitive information. The same problem arises when releasing set-valued data (e.g., shopping and search logs) consisting of a set of records where each record links an individual to his/her set of distinct items. The AOL search data leak in 2006 (Barbaro and Zeller, 2006) is an explicit example that shows the implications of inappropriate anonymization on data privacy. The query logs of 650k individuals were released after omitting all explicit identifiers. They were later withdrawn due to multiple reportings of privacy breaches.

To better illustrate the problem, let us consider a dataset of individuals and their corresponding sets of searched data items. Let us assume that an attacker

---

[1]A person who is intentionally trying to link individuals to their sensitive information

knows that Alice, an individual whose items figure in publicly available dataset, has used two search items: {*SideEffects* and *Vomiting*}. The attacker's background knowledge consists of the set of these two items. If it happens that one and only one record exists in the released dataset where both *Vomiting* and *SideEffects* belong to the same itemset, such as the case of record r_1 in the example shown in Figure 1, the attacker can link the individual Alice to r_1. Here, the attacker, who has partial knowledge of some of the data items searched by the individual, is able to

ID	Records
r_1	{Oncologist, Treatment, Cancer, Surgery, Side Effects, Vomiting}
r_2	{Oncologist, Treatment, Cancer, Surgery, Side Effects, Nausea}
r_3	{Oncologist, Treatment, Cancer}
r_4	{Treatment, Surgery, Side Effects, Chemotherapy}
r_5	{Oncologist, Treatment, Cancer}
r_6	{Oncologist, Surgery}

(a) Original set-valued dataset

ID	Record chunks		Item chunk
	C_1	C_2	CT
r_1	{Oncologist, Treatment, Cancer}	{Surgery, Side Effects}	Vomiting, Nausea, Chemotherapy
r_2	{Oncologist, Treatment, Cancer}	{Surgery, Side Effects}	
r_3	{Oncologist, Treatment, Cancer}		
r_4	{Treatment}	{Surgery, Side Effects}	
r_5	{Oncologist, Treatment, Cancer}		
r_6	{Oncologist}	{Surgery}	

(b) Disassociated dataset with record chunks C_1 and C_2

Figure 1: Example scenario.

determine the complete itemset and links it to the corresponding individual.

There has been significant work in anonymization (Samarati, 2001; Sweeney, 2002; Machanavajjhala et al., 2006; Xiao and Tao, 2006; Dwork et al., 2006; Terrovitis et al., 2012) to cope with this problem. Categorization, on the one hand, separates the data into sensitive and non-sensitive categories of values, which unfortunately can be hard to adopt in real world applications due to the elastic meaning of the sensitivity (Terrovitis et al., 2012). Generalization, on the other hand, transforms values into a general form and creates groups to eliminate possible linking attacks. Despite its efficiency in privacy preservation, generalization has a major drawback related to the loss in data utility (Xiao and Tao, 2006), making it less efficient for aggregate analysis. Anonymization by disassociation (Terrovitis et al., 2008; Loukides et al., 2014a; Loukides et al., 2015) is one recent method that preserves the original items keeping them intact, without applying generalization or substitution. Instead, it uses some form of bucketization and separates the records into clusters and chunks to hide their associations. More specifically, disassociation transforms the original data into k^m-anonymous (Terrovitis et al., 2008) records by dividing the dataset 1) horizontally, creating clusters of similar records and 2) vertically, creating k^m-anonymous chunks of similar sub-records (e.g., record chunks C_1 and C_2 in Figure 1(b)) and an item chunk containing items that appear less than k times.

Disassociation lies under a general category of techniques (Li et al., 2012; Xiao and Tao, 2006; Ciriani et al., 2010) that preserve privacy by dividing the dataset to maximize the utility. It provides strong theoretical guarantees but unfortunately remains vulnerable when the dataset is subject to what we call a *cover problem*, which is defined in Section 5.

In this article, we evaluate the privacy provided by the disassociation of a set-valued dataset and demonstrate that it might be breached whenever this dataset is subject to a *cover problem*. We show that attackers with different levels of partial background knowledge weak, moderate, or strong are able to compromise the privacy of a disassociated dataset. We also show that, as many other bucketization techniques (Xiao and Tao, 2006; Li et al., 2012) in their attempt to protect against attribute disclosure, preserving privacy tends to be overly optimistic (Wong et al., 2011; Kifer, 2009; al Bouna et al., 2015a; al Bouna et al., 2015b).

The contributions of this research work can be summarized as follows:

- We present a formal model of the set-valued data

and the disassociation technique.

- We define the cover problem and investigate its repercussion on the disassociation of the dataset.

- We propose the so-called quantitative privacy breach detection algorithm, and study its efficiency. The proposed algorithm shows to what extent the privacy of a disassociated dataset can be breached.

The remainder of this paper is organized as follows. In the next section, we present prior work in data anonymization and discuss some of the techniques used to breach well known privacy constraints. In Section 3, we give some insights on privacy preserving using the disassociation technique. Section 4 defines the attacker model including the three types of attackers: strong, moderate, and weak. In Section 5, we present the cover problem and the resulting privacy breach in a disassociated dataset. We finally present a set of experiments to evaluate the de-anonymizing algorithm in Section 6.

2 RELATED WORK

Anonymization Techniques have been a topic of intense research in the last decade. Several have been proposed as a way to protect privacy by guaranteeing that the probability of linking an individual to a specific record will not be greater than a certain threshold (e.g., $1/k$ or $1/l$). Some rely on generalization (Samarati, 2001; Sweeney, 2002; Machanavajjhala et al., 2006); replacing some values with more general values based on a given taxonomy or an interval for numerical values, while others on bucketization (Xiao and Tao, 2006; Ciriani et al., 2010; Biskup et al., 2011); separating what is sensitive from non-sensitive. Disassociation lies under this category of bucketization techniques, preserves utility by keeping the values intact without any alteration, but it does not divide the attributes into sensitive and non-sensitive.

Dwork defined the notion of differential privacy (Dwork et al., 2006) to handle private data publishing efficiently. It gained much popularity among computer scientists providing strong assumptions on the way that data should be released. It is based on a strong mathematical foundation and guarantees that an attacker will not be able to learn any information about an individual that he/she already knows if the individual is removed from the dataset. In order to achieve differential privacy, approaches tend to release privatized data by introducing noise to query results. Here, despite the efficiency provided by differ-

ential privacy, we opt for disassociation since it publishes truthful datasets.

Yeye et al. (He and Naughton, 2009) extended *k*-anonymity to set-valued context, in which there is no distinction between quasi-identifying, sensitive or non-sensitive values. In their approach, the authors generate a *k*-anonymous dataset via a top-down partitioning privacy technique based on local generalization. In a similar approach, the authors in (Fard and Wang, 2010) adopted the *k*-anonymity privacy constraint for a transactional dataset. They proposed a clustering based technique to group *similar* transactions into clusters, in order to reduce generalizations and suppression thus reducing information loss. A common problem with these methods is that a large part of the initial items are usually missing from the anonymized data due to generalization or suppression. In (Jia et al., 2014), the authors propose a privacy notion ρ-uncertainty to ensure that there are no sensitive association rules that can be inferred with confidence higher than ρ. Truthful association rules however can still be derived, but this requires, in a similar manner to bucketization techniques, distinction between the sensitive and non-sensitive values.

Preserving privacy by disassociation is a promising technique in which the values are kept intact without any alteration: neither generalization nor suppression. It thus falls under the general category of bucketization techniques but again, without the need to distinguish between sensitive and non-sensitive values.

Attacks on Anonymization focus on the identification of sensitive information, information that is meant to be private, by either exploiting defects in the privacy constraints, a bug in the mechanism that implements it or even by assuming knowledge of the underlying anonymization algorithm such as the minimality attack (Cormode et al., 2010; Wong et al., 2007).

We discuss here some of the interesting attacks that have been reported in the data anonymization literature. Homogeneity attack (Samarati, 2001) is considered successful when there is a lack of diversity among the values of the sensitive attribute. Background knowledge (Machanavajjhala et al., 2006) attack compromises anonymization, when the attacker's background knowledge includes information about the sensitive values of a specific individual or partial knowledge of the distribution of sensitive and non-sensitive values. Unsorted matching attack (Sweeney, 2001) is related to the order in which the tuples appear in the released table that can leak sensitive information if it is preserved in the anonymized dataset.

The attacks mentioned previously present some

Table 1: Notations

$\mathcal{T}$	a table containing individuals related records
r	a record of $\mathcal{T}$; set of items associated with a specific individual of a population
$[r_i]_m$	a set of all m-combinations of items in the record r_i
I	an itemset of $\mathcal{D}$ that might or not be grouped together in a record of $\mathcal{T}$
$R_{\mathcal{T}}$	a set of records in T grouped as a cluster
$s(I,\mathcal{T})$	the number of records in $\mathcal{T}$ that contain I
$\mathcal{T}^*$	a table anonymized using the disassociation technique
C	a chunk in a disassociated dataset; is a set of sub-records of $\mathcal{T}$ in $\mathcal{T}^*$
C_T	an item chunk in a disassociated; dataset a quasi-identifier group
R_C	a record chunk; is a set of sub-records of R in $\mathcal{T}^*$
$\mathcal{B}$	the background knowledge of an attacker

insights on the privacy problems that might be encountered when anonymizing sensitive information in an outsourcing scenario. To the best of our knowledge, none has explored the privacy breach in a disassociated dataset. In this paper, we highlight and evaluate the ability of an attacker to link his/her partial background knowledge represented by the *m* items that he/she is allowed to have, according to the privacy constraint k^m-anonymity, to less than *k* records. The works described in (Wong et al., 2011; Ressel, 1985) are good examples of attacks that follow the same convention by identifying flaws in the anonymization techniques allowing attackers to use their background knowledge to breach privacy or extract foreground knowledge. The authors assume that the correlation between the values in an anonymized dataset could be used to breach the privacy of the anonymization techniques that use *l*-diversity in their underlying privacy mechanism.

3 PRESERVING PRIVACY BY DISASSOCIATION

3.1 Formalization

Let $\mathcal{D} = \{x_1,...,x_d\}$ be a set of items (e.g., supermarket products, query logs, or search keywords). Any subset $I \subseteq \mathcal{D}$ is an itemset (e.g., items searched together). Let $\mathcal{T} = (r_1,...,r_n)$ be a dataset of records, where $r_i \subseteq \mathcal{D}$ for $1 \le i \le n$ is a record in $\mathcal{T}$ and r_i is associated with a specific individual of a population. We use $[r_i]_m$ to denote the set of all m-combinations of the record r_i. Equivalently, $[r_i]_m$ is the set of m items subsets of r_i. Note that $\mathcal{D}$ is no more than the set of items in $\mathcal{T}$: $\mathcal{D} = \bigcup_{i=1}^{n} r_i$.

We use $R_{\mathcal{T}}$ to define a subset of records in $\mathcal{T}$. $s(\mathcal{T})$ and $s(R_{\mathcal{T}})$ denote the numbers of records in $\mathcal{T}$ and $R_{\mathcal{T}}$ respectively and thus, $s(I,\mathcal{T})$ denotes the number of records in $\mathcal{T}$ that contain I.

Table 1 defines the basic concepts and notations used in the paper.

3.2 Disassociation

Disassociation works under the assumption that the items should neither be altered, suppressed, nor generalized, but at the same time the dataset should serve the k^m-anonymity privacy constraint (Terrovitis et al., 2008). k^m-anonymity guarantees that an attacker who knows up to m items, grouped together in an itemset $I \subseteq \mathcal{D}$, cannot associate them with less than k records from $\mathcal{T}$, where $k \geq 2$ is a user-defined constant. Formally, k^m-anonymity is defined as follows:

Definition 1 (k^m-anonymity). *Given a dataset of records $\mathcal{T}$ whose items belong to a given set of items $\mathcal{D}$, we say that $\mathcal{T}$ is k^m-anonymous if $\forall I \subseteq \mathcal{D}$ and $|I| \leq m$, the number of records that contain I in $\mathcal{T}$ is greater than or equal to k, $s(I, \mathcal{T}) \geq k$.*

In a practical scenario, k^m-anonymity cannot be achieved without a severe loss in data utility. The disassociation technique (Terrovitis et al., 2012; Loukides et al., 2014b), for its part, ensures privacy through bucketization to provide better utility when it comes for frequent itemsets discovery and aggregate analysis.

Disassociation separates the dataset into clusters of k^m-anonymous record chunks and an item chunk. The key idea is to sort records based on the most frequent items and then group them horizontally into smaller disjoint clusters $\{R_1, ..., R_n\}$. It partitions, in a next step, the clusters vertically into k^m-anonymous record chunks $\{C_1, ..., C_n\}$ and an item chunk C_T to hide infrequent combinations. The record chunks are created subsequently as long as there are items that can be grouped together in a way to satisfy the k^m-anonymity privacy constraint. The remaining items, the ones that occur less that k times, are moved to the item chunk C_T.[2]

Formally, given a dataset $\mathcal{T}$, applying disassociation on $\mathcal{T}$ will produce an anonymized dataset $\mathcal{T}^*$ composed of n clusters where each is divided into a set of record chunks and an item chunk,

$$\mathcal{T}^* = (\{R_{1_{C_1}}, ..., R_{1_{C_t}}, R_{1_{C_T}}\}, ..., \{R_{n_{C_1}}, ..., R_{n_{C_t}}, R_{n_{C_T}}\})$$

such that $\forall R_{i_{C_j}} \in \mathcal{T}^*$, $R_{i_{C_j}}$ is k^m-anonymous, where,

- $R_{i_{C_j}}$ represents the itemsets of the ith cluster that are contained in the record chunk C_j.

- $R_{i_{C_T}}$ is the item chunk of the ith cluster containing items that occur less than k times.

The example in Figure 1(b) shows that the disassociated dataset contains only one cluster with two

---

[2]The shared chunk as defined in the original paper (Terrovitis et al., 2012) is omitted here for simplicity.

k^m-anonymous record chunks (k=3 and m=2) and an item chunk, thus, $\mathcal{T}^* = (\{R_{1_{C_1}}, R_{1_{C_2}}, R_{1_{C_T}}\})$.

In a disassociated dataset, privacy is preserved by assuming that the record chunks are k^m-anonymous and that the records that are not k^m-anonymous in the original dataset, are k^m-anonymous in *at least* one of the datasets resulting from the inverse transformation of the disassociated dataset. This privacy guarantee is formally expressed as follows:

Definition 2 (Disassociation Guarantee). *Let $\mathcal{G}$ be the inverse transformation of $\mathcal{T}^*$. $\mathcal{G}$ takes an anonymized dataset $\mathcal{G}(\mathcal{T}^*)$ and outputs the set of all possible datasets $\{\mathcal{T}'\}$. $\forall I \subseteq \mathcal{D}$, $|I| \leq m$, $\exists \mathcal{T}' \in \mathcal{G}(\mathcal{T}^*)$ where $s(I, \mathcal{T}') \geq k$.*
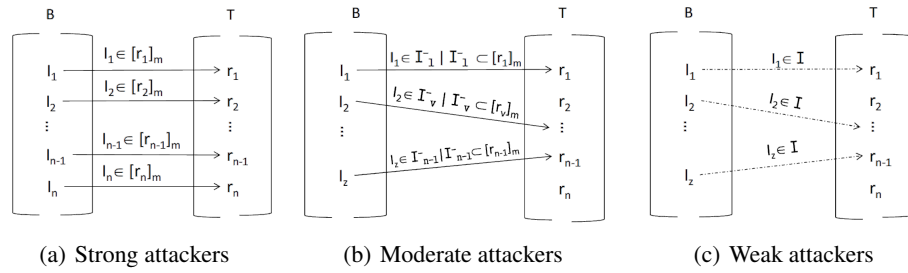
The Disassociation Guarantee ensures that for any individual with a complete record r, and for an attacker who knows up to m items of r, at least one of the datasets reconstructed by the inverse transformation has r hidden by repeating its existence k times.

That is, given m items of r, if r exists less than k times in the original dataset, the attacker will not be able to link back the record to the individual with a probability greater than $1/k$ in the disassociated dataset. This record r, as all other records, exists in at least one of the inverse transformations k times.

4 ATTACKER MODEL

In accordance with the work in (Terrovitis et al., 2012; Loukides et al., 2014b; Terrovitis et al., 2008), we consider that the attacker intends to link individuals to their anonymized records (e.g., complete set of search logs or purchased supermarket products). This can be done, in a trivial anonymization, by tracing unique combinations of items. For example, an attacker who knows that Alice has searched for side effects and vomiting is represented by a background knowledge set of items {*Side Effects, Vomiting*}. Since one and only one such record exists in the dataset of Figure 1(a), the attacker can easily link this record back to Alice. In addition, we assume that the attacker has knowledge of the disassociation technique.

We also assume that the attacker may have background knowledge consisting of at most m items; knowing these m items, he/she will not be able to link them to less than k records in the dataset (i.e., adopting k^m-anonymity, the same privacy guarantee as in (Terrovitis et al., 2012)). Some models add to it negative background knowledge (e.g., the attacker knows that an individual has not posed a specific query, that an item is not purchased by a given individual, etc.). Again, as in (Terrovitis et al., 2012) and many other

(a) Strong attackers   (b) Moderate attackers   (c) Weak attackers

Figure 2: Associations between the background knowledge of attackers based on their category and the records in $\mathcal{T}$.

bucketization techniques (Xiao and Tao, 2006) (Li et al., 2012), we do not assume this kind of negation knowledge. Here, we classify attackers into three categories: strong, moderate, and weak, depending on their level of background knowledge and consequently their ability to associate individuals with their records in $\mathcal{T}$.

Strong Attackers have a background knowledge $\mathcal{B} = \{I_1, ..., I_n\}$ such that $\forall I_i \in \mathcal{B}$, I_i is of size m, $|I_i| = m$ and *well picked*[3] from the set of m-combinations $[r_i]_m$ of a record r_i, $I_i \in [r_i]_m$. For these attackers, as shown in Figure 2(a), there exists a bijective function that links itemsets in $\mathcal{B}$ to records in $\mathcal{T}$.

Moderate Attackers have a background knowledge $\mathcal{B} = \{I_1, ..., I_z\}$, $z < n$ such that $\forall I_i \in \mathcal{B}$, I_i is of size m, $|I_i| = m$ and *chosen at random* from a strict subset I_v^- of the m-combinations $[r_v]_m$ of a record r_v, $I_i \in I_v^- | I_v^- \subset [r_v]_m$. For these attackers, as shown in Figure 2(b), there exists a one-to-one injective function that links itemsets in $\mathcal{B}$ to records in $\mathcal{T}$.

Weak Attackers have a background knowledge $\mathcal{B} = \{I_1, ..., I_z\}$, $z < n$ such that $\forall I_i \in \mathcal{B}$, I_i is of size m, $|I_i| = m$ and *chosen at random* from a set of association rules and frequent itemsets I, $I_i \in I$, that are not necessarily in $\mathcal{D}$. The goal of weak attackers is limited to reconstruct the disassociated dataset to its original form but they are unable to link records to individuals (see Figure 2(c)).

5 PRIVACY EVALUATION IN A DISASSOCIATED DATASET

In this section, we define the cover problem and we demonstrate its repercussion on data privacy. More specifically, we evaluate the privacy breach that might occur in a disassociated dataset due to the cover problem.

---

[3]More details on how these itemsets are picked will be provided in the experiments section

5.1 Cover Problem

A cover problem is defined by the ability to associate one-to-one or one-to-many items in two subsequent record chunks in the disassociated data. The target item from the first chunk has equal or higher support and we call it the covered item. Formally the cover problem is defined as follows:

Definition 3 (Cover Problem). *Given a set of items in $R_{iC_{j-1}}$ $(j \geq 2)$ that have a support greater than or equal to the support of an item $x_{i,j} \in R_{iC_j}$, $I_{i,j-1} = \{y : y \in R_{iC_{j-1}}$ and $s(y, R_{iC_{j-1}}) \geq s(x_{i,j}, R_{iC_j})\}$, we say that a cover problem exists if $\exists y_{i,j-1} \in I_{i,j-1}$ such that $s(y_{i,j-1}, R_{iC_{j-1}})$*

$$= s(I_{i,j-1}, R_{iC_{j-1}}) = \min_{\forall y \in I_{i,j-1}} s(y, R_{iC_{j-1}}).$$

To give an example of the cover problem, consider Figure 3(a). The set of items in the previous record chunk R_{1C_1} having support greater than or equal to e is $I_{1,1} = \{a, b, c\}$. The support of the itemset $I_{1,1}$ in R_{1C_1}, which is $s(I_{1,1}, R_{1C_1})$, is equal to 2. In turn, it is equal to the minimum support of the values in $I_{1,1}$, which, in our example, is $s(c, R_{1C_1})$. Therefore we say that the item c is covered by the items a and b.

Another example can be extracted from Figure 1(b), the item *Cancer* in I_1 is covered by the remainder of the items in I_1. In fact, all of the occurrences of the item *Cancer* are associated with the items *Treatment* and *Oncologist* and therefore, by associating the item *Side Effects* with *Cancer*, *Side Effects* will eventually be linked to both *Treatment* and *Oncologist*.

5.2 Privacy Breach

Recall from Subsection 3.2 and Definition 2, that the disassociation technique hides infrequent records; records that occur less than k times in the original dataset for a given m items, by 1) dividing them into k-anonymous sub-records in record chunks and 2) ensuring that all the records reconstructed by the inverse transformation are k^m-anonymous in at least one of the resulting datasets.

(a) Cover problem example in $\mathcal{T}^*$

(b) Datasets reconstructed by the inverse transformation of $\mathcal{T}^*$

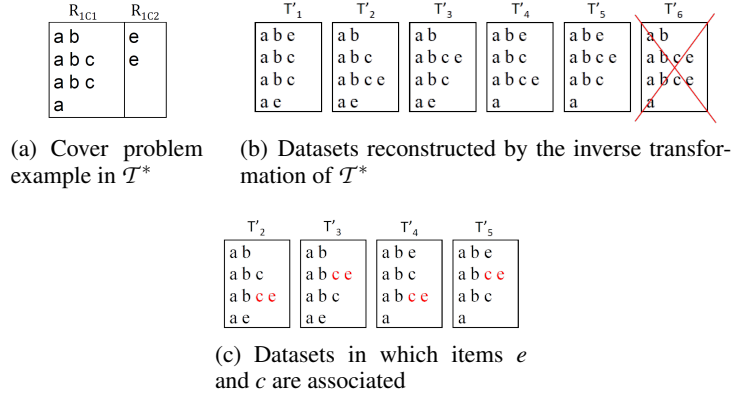(c) Datasets in which items e and c are associated

Figure 3: Cover problem and privacy breach examples.

Intuitively, a privacy breach occurs if an attacker is able to link m items, which he/she already knows about an individual, to less than k records in all the datasets reconstructed by the inverse transformation. More subtle is when these records contain the same set of items in all the reconstructed datasets thus, linking more than m items to the individual or worse leading to a complete *de*-anonymization; linking, with certainty, the complete set of items to the individual. (Figure 3(c) is an example of such a *de*-anonymization. The attacker already knows that an individual has searched for items e and c and thus can relate him/her them to the same items a and b in all the reconstructed datasets).

We will show in the following that this privacy breach might occur whenever the dataset is subject to a cover problem. Formally speaking:

Lemma 1. *Let T^* be a disassociated dataset subject to a cover problem and involving an item $x_{i,j} \in R_{iC_j}$ and a covered item $y_{i,j-1} \in R_{iC_{j-1}}$. We say that a privacy breach might occur if $\{x_{i,j},\ y_{i,j-1}\} \subseteq I$ where $I \in \mathcal{B}$, the attacker's background knowledge.*

We show in what follows that an attacker is able to breach the privacy provided by the disassociation technique, if his/her background knowledge contains the items involved in the association that led to the cover problem.

Proof. The dataset $\mathcal{T}^*$ is subject to a cover problem and therefore, there exists an item $x_{i,j} \in R_{iC_j}$ $(j \geq 2)$ that can be associated with a covered item $y_{i,j-1} \in I_{i,j-1}$ where $I_{i,j-1} = \{y : y \in R_{iC_{j-1}}$ and $s(y, R_{iC_{j-1}}) \geq s(x_{i,j}, R_{iC_j})\}$ is the set of items in $R_{iC_{j-1}}$ having support greater than or equal to the support of $x_{i,j}$.

Let us assume that the items $x_{i,j}$ and $y_{i,j-1}$ are associated together in k records in at least one of the datasets reconstructed by the inverse transformation

of $\mathcal{T}^*$. Since $y_{i,j-1}$ is a covered item, $x_{i,j}$ will also be associated k times with all the items covering $y_{i,j-1}$ in $I_{i,j-1}$. While this is correct from a privacy perspective, it cannot be considered for disassociation. In fact, these items, $x_{i,j}, y_{i,j-1}$ and the covering items in $I_{i,j-1}$ are considered k-anonymous and therefore should have been allot to the record chunk $R_{iC_{j-1}}$ according to disassociation[4].

Now, if the attacker's background knowledge $\mathcal{B}$ consists of itemsets of size m and there exists an itemset I that contains both $x_{i,j}$ and $y_{i,j-1}$, $\{x_{i,j},\ y_{i,j-1}\} \subseteq I$, a privacy breach will occur. In fact, these m items will never be associated together in k records in any of the datasets reconstructed by the inverse transformation of $\mathcal{T}^*$. $\square$

Figure 3(b) describes six possible datasets, $\mathcal{T}'_1$, ...,$\mathcal{T}'_6$, reconstructed by the inverse transformation of the disassociated dataset shown in Figure 3(a). In this example, only five datasets $\mathcal{T}'_1$, ..., $\mathcal{T}'_5$ are valid ones. $\mathcal{T}'_6$ is omitted due to the cover problem and the knowledge of the disassociation algorithm. Given that the item e is associated with the covered item c is $k = 2$ times, consequently e is associated with a and b in k records and thus, according to the disassociation technique, the item e should have remained in the first record chunk R_{1C_1}.

If the attacker's background knowledge contains the items c and b, he/she will be able to link every record containing c and e to less than k records in the record chunk, thus breaching the privacy of disassociation (see Figure 3(c)).

This privacy breach is also detected in the example of Figure 1(b). The item *Cancer* in R_{1C_1}, is one of the least frequent items in the set of items having support greater than or equal to the

---

[4]Vertical partitioning, creation of record chunks, preserves k-anonymous itemsets in the same record chunks.

item *Side Effects* in the record chunk $R_{1_{C_2}}$, $I_1 = \{Oncologist, Cancer, Treatment\}$. *Cancer* is covered by the remainder of the items in I_1. If *Side Effects* is associated with *Cancer* in k records, *Side Effects* will be associated with *Treatment* and *Oncologist* in k records or alternatively saying that the itemset { *Side Effects*, *Cancer*, *Treatment* and *Oncologist* } is k-anonymous. As a consequence, according to the disassociation technique, these items should have belonged to the same record chunk which does not correspond to the disassociated dataset of Figure 1(b).

5.3 Quantitative Privacy Breach Detection Algorithm

We present, in the following, the quantitative privacy breach detection algorithm to evaluate the privacy breach in a disassociated dataset based on the cover problems. The algorithm takes a disassociated dataset $\mathcal{T}^*$, and the attacker's background knowledge $\mathcal{B}$ that includes itemsets of size m and evaluates the privacy breach represented by the number of vulnerable records in the disassociated dataset $\mathcal{T}^*$. It iterates, in Step 3, over each cluster R_i in $\mathcal{T}^*$ in an ascending order to evaluate, for each item $x_{i,j}$ in record chunk $R_{i_{C_j}}$, the number of privacy breaches in all the previous record chunks. To do so, the algorithm 1) retrieves in Step 10 the itemset $I_{i,l}$ of items having support greater than or equal to the support of $x_{i,j}$ and 2) verifies in Step 11 whether $x_{i,j}$ and any item y in $I_{i,l}$ is subject to a cover problem. A privacy breach is noted if both items $x_{i,j}$ and y are included in any of the itemsets I of the attackers' background knowledge $\mathcal{B}$. The algorithm returns the total number of vulnerable records that is computed based on the sum of the maximum number of privacy breaches determined per item.

Although the proposed algorithm covers a broader aspect of the privacy breach by estimating the number of records that are vulnerable due to the cover problem, it can be extended to provide better insights on how many records are *de*-anonymized; linked to the individual with certainty. In fact, as discussed in Subsection 5.2, a privacy breach is found not only when completely *de*-anonymizing a record but also when an attacker is able to link more than m items to the individual. This is noted in the algorithm every time the items involved in the association that led to the cover problem are part of the attacker's knowledge.

Algorithm 1: Quantitative Privacy Breach Detection algorithm.

Require: $\mathcal{T}^*, \mathcal{B}$
Ensure: total
1: $max \leftarrow 0$;
2: $total \leftarrow 0$;
3: **for** each $R_i \in \mathcal{T}^*$ **do**
4:     get j, the number of record chunks in R_i;
5:     **for** each $R_{i_{C_j}} \in R_i$ **do**
6:         $breach \leftarrow 0$;
7:         $l \leftarrow j - 1$;
8:         **for** each item $x_{i,j} \in R_{i_{C_j}}$ **do**
9:             **while** ($l \geq 0$) **do**
10:                 get $I_{i,l}$;
/**$I_{i,l} = \{y : y \in R_{i_{C_l}}$ and $s(y, R_{i_{C_l}}) \geq s(x_{i,j}, R_{i_{C_j}})\}$*/
11:                     **if** $s(I_{i,l}, R_{i_{C_l}}) = \min_{\forall y \in I_{i,l}} s(y, R_{i_{C_l}})$ and $\{y_{i,l}, x_{i,j}\} \subseteq I$ **then**
/**$I \in \mathcal{B}$*/
12:                       $breach + +$;
13:                 **end if**
14:                 $l - -$;
15:             **end while**
16:         **end for**
17:         **if** $breach > max$ **then**
18:             $max \leftarrow breach$;
19:         **end if**
20:     **end for**
21:     $total \leftarrow total + max$;
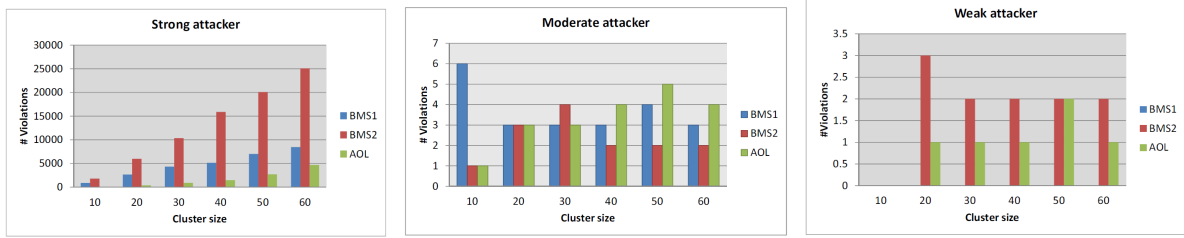22: **end for**
23: **return** $total$;

---

6 EXPERIMENTS

We conducted a series of experiments on real datasets described in Table 2, the click-stream data, BMS-WebView-1 and BMS-WebView-2, and the AOL search query logs. We implemented all methods in JAVA and performed the experiments on an Intel Core i7-4702MQ CPU at 2.2GHz with 8GB of RAM. The goal of these experiments is to:

- Evaluate the privacy breach represented by the number of vulnerable records using the quantitative privacy breach detection algorithm on the aforementioned datasets with various attackers: strong, moderate, and weak,

- Evaluate the performance of the quantitative privacy breach detection algorithm.

Table 2: Datasets properties.

Dataset	# of distinct individuals	# of distinct items
AOL	65517	1216655
BMS-WebView-1	59602	497
BMS-WebView-2	77512	3340

The tests were performed using three types of attackers, strong, moderate and weak, where each of

(a) Strong attacker privacy breach evaluation

(b) Moderate attacker privacy breach evaluation

(c) Weak attacker privacy breach evaluation

Figure 4: Privacy breach and performance evaluations.

which has a background knowledge simulated as follows:

Strong Attacker: the background knowledge of this attacker consists of itemsets of size 2 ($m = 2$) picked from the 2-combinations of each and every record in the dataset. Since it is a strong attacker, we assume that these itemsets contain the items that are involved in the association that will lead to a cover problem (i.e., well picked). Therefore, a privacy breach occurs whenever a cover problem is found.

Moderate Attacker: the background knowledge of this attacker consists of itemsets of size 2 ($m = 2$) chosen at random from the 2-combinations of a strict subset of $\mathcal{D}$ (records in $\mathcal{T}$).

Weak Attacker: to simulate the background knowledge of this attacker, we have chosen at random 10 items from the dataset (for each test) and another 10 items from wordnet (Miller, 1995) and generated their 2-combinations. This gave us a background knowledge $\mathcal{B}$ of size 190.

Privacy Breach Evaluation. In this test, we evaluate the privacy breach caused by the cover problem using the quantitative privacy breach detection algorithm. Three test cases were studied. In each of the test cases, the algorithm evaluates the aforementioned datasets by varying the maximum cluster size with fixed $k = 3$, $m = 2$ according to the attacker's background knowledge. The results are shown in Figure 4. It is not surprising that for strong attackers the privacy breach is most explicit. In fact, strong attackers' background knowledge, consisting of all the itemsets of size $m = 2$, are linked to each of the individuals in the dataset. Therefore, whenever a cover problem is found, a privacy breach is noted. In addition, we can notice that for the strong attacker as exhibited in the three datasets, the privacy breach increases linearly with the cluster size. Larger clusters include more cover problems and thus more privacy
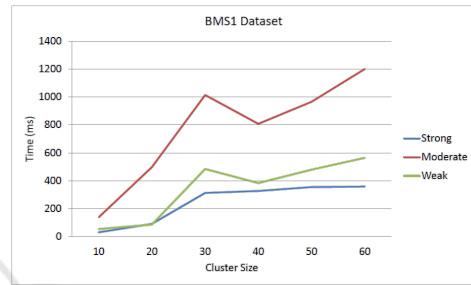

Figure 5: Quantitative privacy breach detection algorithm performance evaluation.

breaches. This is not reflected by moderate and weak attackers. These attackers have background knowledge generated from a strict subset of randomly chosen items from the dataset and other items from external sources. Moreover and unlike the strong attackers' background knowledge, moderate and weak attackers' background knowledge are linked to a subset of records in the datasets. This fact justifies the low number of violations found in the dataset for these attackers.

Performance Evaluation. Here, we evaluate the performance of the quantitative privacy breach detection algorithm on the BMS-WebView-1 dataset for the three types of attackers; strong, moderate and weak. The results shown in Figure 5 demonstrate that the algorithm performs in a generally stable fashion when increasing the cluster size. The best performance is obtained when dealing with strong attackers. The algorithm treats every cover problem as a privacy breach and therefore, unlike for moderate and weak attackers, there is no need to search for the items that are involved in the association that led to a cover problem in the background knowledge of the attackers.

7 CONCLUSION

In this article, we have proposed a formal model of the set-valued data and the disassociation technique. The cover problem has been defined, while its effects on the disassociation of the dataset has been regarded. Such investigations have led to the quantitative privacy breach detection algorithm, whose efficiency has been studied. By this way, we have shown in what extent a disassociated dataset can be vulnerable. In the near future, we intend to develop partial suppression in disassociated dataset and evaluate the gain in utility that the disassociation provides.

REFERENCES

al Bouna, B., Clifton, C., and Malluhi, Q. M. (2015a). Anonymizing transactional datasets. *Journal of Computer Security*, 23(1):89–106.

al Bouna, B., Clifton, C., and Malluhi, Q. M. (2015b). Efficient sanitization of unsafe data correlations. In *Proceedings of the Workshops of the EDBT/ICDT 2015 Joint Conference (EDBT/ICDT), Brussels, Belgium, March 27th, 2015.*, pages 278–285.

Barbaro, M. and Zeller, T. (2006). A face is exposed for aol searcher no. 4417749.

Biskup, J., PreuB, M., and Wiese, L. (2011). On the inference-proofness of database fragmentation satisfying confidentiality constraints. In *Proceedings of the 14th Information Security Conference*, Xian, China.

Ciriani, V., Vimercati, S. D. C. D., Foresti, S., Jajodia, S., Paraboschi, S., and Samarati, P. (2010). Combining fragmentation and encryption to protect privacy in data storage. *ACM Trans. Inf. Syst. Secur.*, 13:22:1–22:33.

Cormode, G., Li, N., Li, T., and Srivastava, D. (2010). Minimizing minimality and maximizing utility: Analyzing method-based attacks on anonymized data. In *Proceedings of the VLDB Endowment*, volume 3, pages 1045–1056.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 265–284, Berlin, Heidelberg. Springer-Verlag.

Fard, A. M. and Wang, K. (2010). An effective clustering approach to web query log anonymization. In *Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on*, pages 1–11. IEEE.

He, Y. and Naughton, J. F. (2009). Anonymization of set-valued data via top-down, local generalization. *Proc. VLDB Endow.*, 2(1):934–945.

Jia, X., Pan, C., Xu, X., Zhu, K., and Lo, E. (2014). -uncertainty anonymization by partial suppression. In Bhowmick, S., Dyreson, C., Jensen, C., Lee, M., Muliantara, A., and Thalheim, B., editors, *Database Systems for Advanced Applications*, volume 8422 of *Lecture Notes in Computer Science*, pages 188–202. Springer International Publishing.

Kifer, D. (2009). Attacks on privacy and definetti's theorem. In *SIGMOD Conference*, pages 127–138.

Li, T., Li, N., Zhang, J., and Molloy, I. (2012). Slicing: A new approach for privacy preserving data publishing. *IEEE Trans. Knowl. Data Eng.*, 24(3):561–574.

Loukides, G., Liagouris, J., Gkoulalas-Divanis, A., and Terrovitis, M. (2014a). Disassociation for electronic health record privacy. *Journal of Biomedical Informatics*, 50:46–61.

Loukides, G., Liagouris, J., Gkoulalas-Divanis, A., and Terrovitis, M. (2014b). Disassociation for electronic health record privacy. *Journal of Biomedical Informatics*, 50(0):46 – 61. Special Issue on Informatics Methods in Medical Privacy.

Loukides, G., Liagouris, J., Gkoulalas-Divanis, A., and Terrovitis, M. (2015). Utility-constrained electronic health record data publishing through generalization and disassociation. In Gkoulalas-Divanis, A. and Loukides, G., editors, *Medical Data Privacy Handbook*, pages 149–177. Springer International Publishing.

Machanavajjhala, A., Gehrke, J., Kifer, D., and Venkitasubramaniam, M. (2006). *l*-diversity: Privacy beyond *k*-anonymity. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, Atlanta Georgia.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Ressel, P. (1985). De Finetti-type theorems: an analytical approach. *Ann. Probab.*, 13(3):898–922.

Samarati, P. (2001). Protecting respondents' identities in microdata release. *IEEE Trans. Knowl. Data Eng.*, 13(6):1010–1027.

Sweeney, L. (2001). Computational disclosure control - a primer on data privacy protection. Technical report, Massachusetts Institute of Technology.

Sweeney, L. (2002). k-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570.

Terrovitis, M., Mamoulis, N., and Kalnis, P. (2008). Privacy-preserving anonymization of set-valued data. *PVLDB*, 1(1):115–125.

Terrovitis, M., Mamoulis, N., Liagouris, J., and Skiadopoulos, S. (2012). Privacy preservation by disassociation. *Proc. VLDB Endow.*, 5(10):944–955.

Wong, R. C.-W., Fu, A. W.-C., Wang, K., and Pei, J. (2007). Minimality attack in privacy preserving data publishing. In *VLDB*, pages 543–554.

Wong, R. C.-W., Fu, A. W.-C., Wang, K., Yu, P. S., and Pei, J. (2011). Can the utility of anonymized data be used for privacy breaches? *ACM Trans. Knowl. Discov. Data*, 5(3):16:1–16:24.

Xiao, X. and Tao, Y. (2006). Anatomy: Simple and effective privacy preservation. In *Proceedings of 32nd International Conference on Very Large Data Bases (VLDB 2006)*, Seoul, Korea.