An Open Source Tool for Game Theoretic Health Data De-Identification

Fabian Prasser, PhD[1], James Gaupp, MS[2], Zhiyu Wan[2], Weiyi Xia, PhD[2], Yevgeniy Vorobeychik, PhD[2], Murat Kantarcioglu, PhD[3], Klaus Kuhn, MD[1], and Brad Malin, PhD[2]
[1]Technical University of Munich, Munich, Germany; [2]Vanderbilt University, Nashville, Tennessee, USA; [3]University of Texas at Dallas, Richardson, Texas, USA

Abstract

Biomedical data continues to grow in quantity and quality, creating new opportunities for research and data-driven applications. To realize these activities at scale, data must be shared beyond its initial point of collection. To maintain privacy, healthcare organizations often de-identify data, but they assume worst-case adversaries, inducing high levels of data corruption. Recently, game theory has been proposed to account for the incentives of data publishers and recipients (who attempt to re-identify patients), but this perspective has been more hypothetical than practical. In this paper, we report on a new game theoretic data publication strategy and its integration into the open source software ARX. We evaluate our implementation with an analysis on the relationship between data transformation, utility, and efficiency for over 30,000 demographic records drawn from the U.S. Census Bureau. The results indicate that our implementation is scalable and can be combined with various data privacy risk and quality measures.

Introduction

The healthcare community is increasingly driven by programs that collect and process large quantities of patient-specific data[1,2]. At the same time, these programs make use of highly detailed aspects about a patient and their daily activities within, as well as outside of, traditional clinical environments. To ensure that research studies are conducted, and healthcare applications are managed, at scale, it is critical to share data beyond the confines of where it was initially collected[3]. The push to make data accessible is exacerbated by initiatives that aim to enhance transparency in activities, reproducability of research findings, and reuse of data for novel investigations. Such initiatives are being driven by federal agencies in the United States (U.S.), such as through the various data sharing policies of the National Institutes of Health[4] and the National Science Foundation[5], as well as in Europe, such as through Policy 0070 of the European Medicines Agency (EMA) for the public dissemination of clinical trials data[6]. However, as the quantity and quality of biomedical data grows, so too does its attractiveness to would-be attackers. There is, for instance, ample evidence to show that data breaches for healthcare organizations have grown substantially over the past several years[7].

Privacy is a complex concept with ethical, legal and societal aspects[8] and, consequently, it requires a mix of social (e.g., consent and trust) and technical constructs to realize[9]. In this work, we confine ourselves solely in the technical space, where a patient's privacy is typically supported through a process referred to as de-identification in the U.S. (the convention we use in this paper) and anonymization elsewhere. De-identification is often achieved through the amendment of data that pertains to an individual's identity. Such amendments may be realized through randomization, generalization to less specific terms, or suppression (i.e., redaction) of factors that can be leveraged to ascertain an individual's identity (e.g., demographic factors). Specific rules for doing so have been codified in laws and regulations[10,11].

In parallel with these developments, a variety of studies have shown that de-identified health data can be re-identified (i.e., linked to identified individuals)[12]. This has sparked a debate in which it has been argued that de-identification fails to provide adequate privacy protection[13]. However, it is important to realize that re-identification studies tend to focus on how attacks can be carried out, not on the likelihood they will be realized. As such, they demonstrate what is possible and not necessarily what is probable[14]. It has long been understood that de-identification involves trading privacy risks off against data quality and that privacy risks can never be entirely eliminated[9].

Most recently, it was suggested that de-identification frameworks should formalize the capabilities of adversaries[15]. This shifts the view to an economic perspective, where the publishers gain benefit from sharing data at a certain level of fidelity, while attackers (i.e., the recipients) gain benefit from re-identifying the data. By modeling re-identification as a Stackelberg (or what is also called a leader-follower) game[16] between two players, it was shown that attacks initially thought to be quite detrimental to the management of biomedical research infrastructure may actually not be likely to occur in practice. Moreover, it was shown that the publisher's gain can be maximized by de-identifying data under the assumption that the adversary will only attempt re-identification if there is a tangible economic benefit[16]. While this approach is a notable advance in the field[17], it has not been made accessible through an easy-to-use software system.

Objective and Contributions

Traditionally, de-identification has been understood to be an optimization problem, in which data is transformed while an increase in privacy protection is traded off against a decrease in data quality. To balance both aspects, models for quantifying them are needed. Typically, a threshold is defined for privacy risks, which reduces the task to a simpler optimization problem, where the objective is to ensure that risk thresholds are met while data quality is maximized. At first glance, the game theoretic model appears to differ from this perspective. This paper reports on an in-depth analysis to identify integration options, as well as how we adapted and transitioned the game theoretic approach into scalable software. We extended ARX, an open source de-identification tool[18], with methods for de-identifying demographic and clinical data under the Stackelberg setting.

The specific contributions of this paper are as follows. First, we present an analysis of the game theoretic approach. Our findings show that (1) a specific variant of the game can be implemented with traditional privacy models and (2) the model can be interpreted in a manner that is consistent with the traditional perspective (i.e., as a combination of a privacy model and a data quality model). This interpretation forms the basis of our integration. Second, we further developed a search space pruning strategy that is specific to the game theoretic model and improves the scalability of the implementation significantly. Third, we describe how we integrated the game theoretic model into ARX (version 3.5.0), such that the implementation is fully compatible with all other de-identification methods implemented by the software. This is notable because it implies that the game theoretic approach can be combined, as well as compared, with other models for quantifying health data privacy risks and quality.

Beyond a software implementation, we report on an extensive experimental evaluation with over 30,000 demographic records from the U.S. Census. The results indicate that our highly scalable implementation often provides an overall payout (a measure that combines risk and utility) to the publisher that is comparable to what can be achieved using more sophisticated, though computationally costly, routines. Finally, we conclude this paper by reporting on ways to further extend our implementation towards the handling of high-dimensional (e.g., genomic) data.

Background and Related Work

Data De-identification

The Privacy Rule of the Health Insurance Portability and Accountability Act (HIPAA)[10] provides specific guidance for the de-identification of health data via the *Safe Harbor* method. This method specifies 18 rules for modifying explicit (e.g., patient names and Social Security Numbers) and quasi-identifying (e.g., dates, ZIP codes with fidelity above 3 digits) factors that constitute sufficiently high re-identification risk. As an alternative, the HIPAA Privacy Rule permits the use of formal risk assessments under its *Expert Determination* methodology[10]. This approach designates health data as de-identified when it is shown that the risk an anticipated recipient of the data could uniquely identify an individual is small. The method is notable as it allows for the application of an explicit adversarial model (i.e., anticipated recipient) and can explicitly manage risk through a quantified mechanism. Moreover, the approach is similar to how privacy is maintained in other countries. For example, the upcoming European General Data Protection Regulation (GDPR) states that ”account should be taken of all the means reasonably likely to be used [for re-identification]”[11].

Data De-identification Software

There are various mature software solutions for facilitating data privacy risk assessments and de-identification, which have proven themselves in practice. μ-ARGUS and sdcMicro are open-source tools developed in the context of official statistics[19]. By contrast, Privacy Analytics Eclipse is a commercial big data platform for risk-based de-identification of structured health records[20]. In this paper, we focus on *ARX*, an open source tool that was also specifically designed for de-identifying biomedical data[18]. ARX has been under constant development since 2011 and has found notable adoption due to its comprehensive feature set and its easy-to-use graphical user interface. For example, it has been recommended by the EMA for implementing Policy 0070[21]. ARX supports a wide variety of models for quantifying privacy risks and data quality as well as multiple models for transforming data.

Data Transformation

Formal de-identification is typically applied to data derived from the healthcare setting by reducing the distinguishability of patient-level records. There are varying approaches for doing so, such as the injection of noise to adhere to emerging privacy models like differential privacy[22]. However, the most popular approach in this domain remains reducing the fidelity of attributes that are likely to be exploited in re-identification attacks through linkage to named individuals in some external resource. This process renders records of a dataset to be less distinguishable for an attacker, thus reducing the certainty in executing a re-identification attack.

A generalization hierarchy is often relied upon for managing the valid transformations that can be applied to patient-level values. Two examples are shown in Figure 1. Here, values of an age attribute are transformed into intervals with decreasing precision over increasing *levels* of generalization. Note that assigning a value to level 0 of its hierarchy leaves the value unchanged. In ARX, generalization hierarchies can be composed for categorical and continuous variables. In the latter case, this is accomplished by specifying functions for performing on-the-fly categorization of the space (e.g., creating an arbitrary grouping of heights or weights).

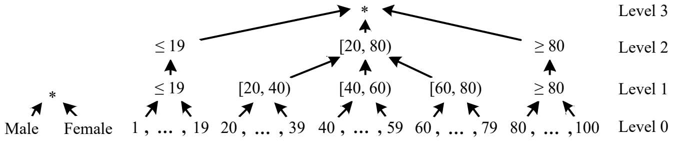
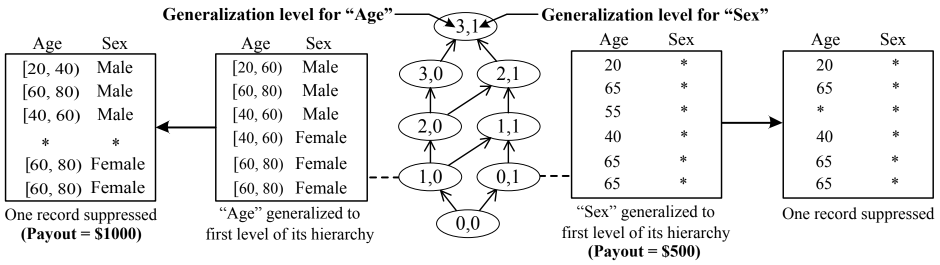


Figure 1: Example generalization hierarchies for patient demographics that are vulnerable to re-identification attacks.

While generalization can be realized in a variety of ways, in this article, we focus on two in particlar. In *full-domain generalization*, all values of an attribute are transformed to the same generalization level in all records. By contrast, in *record-level generalization*, the values in each record can be generalized to different levels. To prevent overgeneralization, the former approach is often combined with subsequent *record-level suppression*. In this process, outliers (e.g., records with a high re-identification risk) are dropped from the dataset. The set of all possible combinations of generalization levels for all attributes forms a *generalization lattice*, where each element is called a *de-identification policy*. The generalization lattice for the example hierarchies from Figure 1 is shown in Figure 2. The latter figure also depicts the results of applying two de-identification policies for full-domain generalization, followed by subsequent record suppression. The payout indicated in the figure will be used as an example throughout this paper.


Generalization level for "Age"
Generalization level for "Sex"
3,1
3,0
2,1
2,0
1,1
1,0
0,1
0,0

Age	Sex
[20, 40)	Male
[60, 80)	Male
[40, 60)	Male
*	*
[60, 80)	Female
[60, 80)	Female

One record suppressed
(Payout = $1000)

Age	Sex
[20, 60)	Male
[60, 80)	Male
[40, 60)	Male
[40, 60)	Female
[60, 80)	Female
[60, 80)	Female

"Age" generalized to first level of its hierarchy

Age	Sex
20	*
65	*
55	*
40	*
65	*
65	*

"Sex" generalized to first level of its hierarchy
(Payout = $500)

Age	Sex
20	*
65	*
*	*
40	*
65	*
65	*

One record suppressed

Figure 2: A generalization lattice, de-identification policies, and output datasets for full-domain generalization.

Models for Quantifying Data Quality and Data Protection

Data transformation can influence the specificity of the data, so it is critical to balance an increase in protection with a decrease in data quality. To do so, we need to define formal measures for both criteria.

As alluded to, there are many models for measuring data protection, but in this work we focus on re-identification risk, because this is the primary concern addressed by current law and regulation. The re-identification risk of a record is typically estimated by calculating the inverse of the size of the group of indistinguishable records to which it belongs. The universe of records that needs to be considered when forming groups depends on assumptions about the adversary's prior knowledge.

First, in the *Prosecutor* model, it is assumed that the adversary already knows a record for a targeted individual is contained in the dataset. As such, groups must be formed using all records from the *output* dataset. This is the risk model underlying the k-anonymity privacy model[23]. Second, in the *Journalist* model, it is assumed the adversary has no prior knowledge about membership. Thus, groups can be formed using a *population table* (i.e., a dataset containing records about *all* individuals from the population) generalized in the same way as the given record. This is the approach underlying the k-map privacy model[24]. If no population table is available, a conservative *estimate* of the size of a record's population group can be derived from the given dataset generalized in the same way as the record.

There are also various models for quantifying data quality. In this article we focus on the entropy-based model proposed by Wan et al.[16]. For a given record r, the information loss function $\mathit{IL}(r)$ returns a number in range $[0,1]$ where a value of 0 indicates that the record has been preserved in its original state and a value of 1 indicates that all data has been suppressed.

The Game Theoretic Model

The game theoretic approach uses models for privacy protection and data quality to trade both aspects off against each other. For this purpose, it incorporates four intuitive parameters to construct a Stackelberg game in which the players, the data publisher (or defender) and the recipient (or attacker), are motivated by monetary incentives:

- **Adversary Gain**: The benefit that the adversary gains for a successfully re-identified record.
- **Adversary Cost**: The adversary's cost to launch a re-identification attack against one record.
- **Publisher Benefit**: The benefit that the publisher receives by sharing a record in its original form.
- **Publisher Loss**: The publisher's loss for one record due to successful re-identification.

By applying a model for quantifying re-identification risks to estimate the *attacker's success probability* $\mathit{SP}(r)$ when attacking a record r, a monetary cost-benefit analysis can be performed. Our implementation supports all models described in the previous section.

The adversary attacks a record r when the expected payoff is positive. More formally:

$$\mathit{Attack}(r) = \begin{cases} 1, & \text{if } \mathit{SP}(r) \cdot \mathit{Gain} > \mathit{Cost} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

As noted earlier, the value of a published record decreases when the amount of information content decreases. Thus, the publisher's payoff for publishing a record r can therefore be defined as:

$$\mathit{Benefit}(r) = (1 - \mathit{IL}(r)) \cdot \mathit{Benefit} \tag{2}$$

Additionally, when a record is successfully attacked, the publisher loses money. The expected loss per record is relative to the success probability of the adversary. Consequently, the publisher's expected overall payout for a record r is:

$$\mathit{Payout}(r) = \mathit{Benefit}(r) - \mathit{Attack}(r) \cdot \mathit{SP}(r) \cdot \mathit{Loss} \tag{3}$$

Methods

System Design

The de-identification methods supported by ARX are centered on generic data transformation algorithms that invoke full-domain generalization followed by record suppression[25]. The algorithms rely on a user-specified 1) privacy model to determine which records need to be suppressed and 2) data quality model for optimizing the output.

In essence, ARX iterates over all available de-identification policies. For each policy, it follows a three step process (ignoring the various implemented optimizations): 1) *Transform* all records according to the policy, 2) *Suppress* records as indicated by the privacy model, and 3) *Assess* data quality using the quality model. Once all of the policies have been processed, ARX selects the policy with the best data quality to de-identify the dataset.

Implementation of the Basic Game

Prior work in game theoretic de-identification pursued a record-level perspective by transforming each record in such a way that the resulting payout is maximized[16]. However, the implementation of the model in ARX relies upon full-domain generalization, as this approach has been shown to be more desirable to the biostatistics community[24]. Later on, we will show that this basic implementation can be used to implement the record-level method.

To integrate the game theoretic model with ARX, we defined a privacy model and a quality model. The privacy model, called *Profitability*, is defined as follows:

$$\mathit{Profitable}(r) = \begin{cases} 1 \text{ if } \mathit{Payout}(r) >= 0 \\ 0 \text{ otherwise} \end{cases} \tag{4}$$

ARX uses this model to decide which records should be suppressed. We note that for any given dataset D, suppressing all records $r \in D$ with $\mathit{Profitable}(r) = 0$ is the optimal strategy for record-level suppression. This is because the decision of whether or not a record should be suppressed can be made *independently* for each record. Specifically, suppressing a record will not affect the information loss measured by the function *IL* for any other record[16]. Moreover, when using a population table or the input dataset to calculate the adversary's success probability *SP*, removing a record from the output will also not affect the value returned by the function *SP* for other records. When using the output dataset to evaluate *SP*, suppressing a nonprofitable record will only increase the value of *SP* for records that are in the same group. However, the group only contains nonprofitable records and increasing the value returned by *SP* makes publishing them even less profitable. As a consequence, suppressing a record does not affect whether publishing other records is profitable to the publisher. Obviously, the optimal strategy is to keep all profitable records and to remove all nonprofitable records.

The quality model is an objective function that measures the overall payout of the publisher for the complete dataset D. Suppressed records will not be published, so we neither gain nor lose any money from them:

$$\mathit{Payout}(D) = \sum_{r \in D} \mathit{Payout}(r) \cdot \mathit{Profitable}(r) \tag{5}$$

By parameterizing ARX with these models, we can solve the game using full-domain attribute generalization followed by record-level suppression. The process is sketched in Figure 3. As can be seen, ARX will suppress all records for which publishing will result in a negative payoff for the publisher. The overall payout for a (potentially generalized) dataset is defined as the sum of the payoffs for each record. The payoff for a suppressed record is defined as zero.

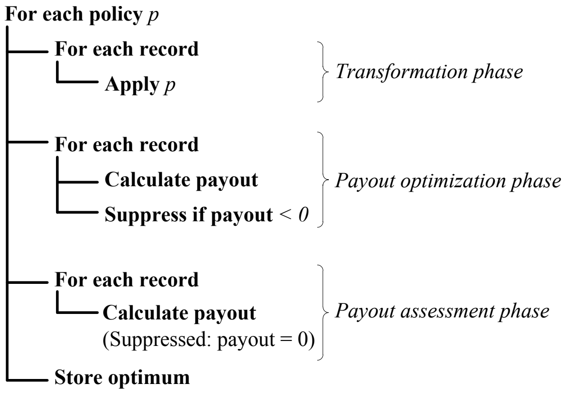


Figure 3: Basic algorithm executed by ARX for solving the game.

The output of this algorithm is globally optimal (in terms of overall publisher payout) regarding all possible transformations that rely on a combination of full-domain generalization and record suppression. This is because the algorithm implements an *exhaustive search*; i.e., it assesses every de-identification policy. For each policy, it first generalizes the dataset, then performs record suppression and stores the resulting payout. Since there is only a single optimal record suppression strategy for any dataset, the algorithm has checked all potentially optimal outputs and the solution must therefore be globally optimal.

We also implemented a pruning strategy that eliminates solution candidates by reducing the search space of potentially optimal policies. To do so, we take advantage of the fact that the entropy-based information loss model used by the game theoretic approach is monotonic over paths in the generalization lattice. This means that, when data is *only* transformed with full-domain generalization, information loss will increase monotonically on each path from the bottom node of the generalization lattice to the top node. We note that this is *not* the case when data is transformed with full-domain generalization followed by record suppression[26]. However, the monotonicity of the quality model can still be invoked to construct a pruning strategy.

For this purpose, we first calculate an upper bound on the maximal payout that can be obtained for a dataset D by assuming that the adversary never attacks (i.e., that $\forall_r\ \mathit{SP}(r) = 0$):

$$\mathit{MaximalPayout}(D) = \sum_{r \in D} \mathit{Benefit}(r) = \sum_{r \in D} (1 - \mathit{IL}(r)) \cdot \mathit{Benefit} \tag{6}$$

Due to the fact that $\mathit{MaximalPayout}(D)$ only depends on $\mathit{IL}(r)$, it follows that it also decreases monotonically within the lattice when data is only transformed with full-domain generalization. We can exploit this to prune portions of the search space by implementing the following strategy.

While traversing the solution space, we always keep track of the *OptimalPayout*, i.e. the payout of the best solution found so far. We note that *OptimalPayout* represents the highest payout obtained using full-domain generalization *and* record suppression. When processing a new policy p, we retrieve a dataset D as a result of the transformation phase. Next, we check whether $\mathit{MaximalPayout}(D) \leq \mathit{OptimalPayout}$. If this is the case, p and all of its successors can be pruned, because the maximal payout that can be obtained by using them is already lower than the current optimum. Our implementation uses the generic framework provided by ARX for implementing such pruning strategies[26].

For context, Figure 2 provides an example. Let us assume that the algorithm first processes the policy $(1, 0)$ by generalizing the data and then suppressing one record. The resulting payout is the current optimum, so *OptimalPayout*=\$1000. Next, the algorithm processes policy $(0, 1)$. In doing so, it first generalizes to retrieve a dataset D' and finds that $\mathit{MaximalPayout}(D') = \$800$. Policy $(0, 1)$ and all of its successors (i.e. $(1, 1)$, $(2, 1)$, $(3, 1)$) can now be pruned because none can result in output with a higher payout than the current optimum ($\$800 \leq \$1000$).

Implementation of Variants of the Game

There are several variants of the game theoretic model that have been proposed[16] that we chose to integrate into ARX.

The first is the *SH-Friendly* variant, which guarantees that the degree of protection from re-identification is no lower than when using the HIPAA Safe Harbor policy. This can be achieved with ARX by using generalization hierarchies that reflect the Safe Harbor policy and by defining appropriate minimal levels of generalization.

The second is the *No-Attack* variant of the game, which guarantees that the adversary has no incentives to ever attack. This variant can be implemented in an optimized manner, as can be seen as follows. Let k be the size of the group of indistinguishable records used for calculating $SP(r)$. As described above, the adversary will only attack a record if:

$$\mathit{SP}(r) \cdot \mathit{Gain} > \mathit{Cost}. \tag{7}$$

So, it follows that:

$$\mathit{SP}(r) > \frac{\mathit{Cost}}{\mathit{Gain}} \;\Rightarrow\; \frac{1}{k} > \frac{\mathit{Cost}}{\mathit{Gain}} \;\Rightarrow\; k > \frac{\mathit{Gain}}{\mathit{Cost}}. \tag{8}$$

This indicates that the *No-Attack* variant of the game implies a threshold on k. Depending on how the game has been configured to quantify re-identification risks (see Section *Models for Quantifying Data Quality and Data Protection*) this can be implemented with k-anonymity[23] or k-map[24]. As an example, given the realistic parameters derived by Wan et al. (i.e. *Cost* = \$4 and *Gain* = \$300[16]), we obtain $k > \frac{300}{4} = 75$.

In addition to enforcing a threshold on k, we also need to maximize the publisher's payout. Since we know that the adversary will never attack, Equation 3 can be simplified as follows:

$$\mathit{Payout}(r) = \mathit{Benefit}(r) - \underbrace{\mathit{Attack}(r)}_{=0} \cdot \mathit{SP}(r) \cdot \mathit{Loss} = \mathit{Benefit}(r) = (1 - \mathit{IL}(r)) \cdot \mathit{Benefit} \tag{9}$$

These specifics of the *No-Attack* variant are important from an implementation perspective. They show that it is sufficient to minimize information loss in order to maximize publisher payout and that payout is monotonic in the lattice when only full-domain generalization is being used[23,24]. This can be used to speed up record-level generalization.

Implementation of Record-Level Generalization

Our implementation of the game theoretic model is fully integrated into ARX. Figure 4 depicts two examples of how the API can be invoked. These illustrate how to de-identify an example dataset D using the game theoretic approach with different models for transforming data and quantifying re-identification risks.

Algorithm 1 Full-domain generalization + record suppression

```
arx ← new ARX();
arx.setPrivacyModel(new Profitability());
arx.setQualityModel(new PublisherPayout());
output ← arx.process(D);
```

Algorithm 2 Record-level generalization

```
for (0 ≤ i < |D|) {
   arx ← new ARX();
   arx.setPrivacyModel(new Profitability());
   arx.setQualityModel(new PublisherPayout());
   arx.setSample({i});
   output ← output ∪ arx.process(D);
}
```

Figure 4: Implementing different transformation models using the ARX API.

The first example shows pseudocode for invoking the API using full-domain generalization followed by record suppression and the prosecutor risk model. It should be noted that ARX can also be configured to use a population table. To do so, the dataset to be de-identified needs to be defined as a *sample* of the population table. With this configuration, our implementation uses the journalist risk model.

In the second example, we invoke the method without an explicit population table to estimate journalist risks as described previously. For this purpose, we de-identify the dataset $|D|$ times, using a sample of exactly one record of D in each execution. The result for D is simply the union of the individual outputs for each record.

Experimental Design

Materials. We evaluate our methods on the extract of 32,561 records from the 1994 U.S. Census and the generalization hierarchies used by Wan et al.[16]. The dataset was enriched with demographics extracted from additional U.S. Census data for the state of Tennessee to enable comparisons with HIPAA Safe Harbor. We used traditional demographics in the form of 1) age, 2) sex, 3) ZIP code, and 4) race as quasi-identifying variables given their known contribution to re-identification attacks.

Parameterization. As a base case, we used the setup from Wan et al.[16]: Adversary cost = \$4, based on the costs for obtaining detailed information about individuals online. Publisher benefit = \$1,200, based on an analysis of grant funding received divided by the number of records published by the five member institutions of the Electronic Medical Records and Genomics (EMERGE) network, an NIH consortium. Publisher loss = \$300, based on an analysis of HIPAA breach violation cases reported by the U.S. Department of Health and Human Services.

In our experiments, we rely on the default parameterization for adversary cost and publisher benefit, but we further defined adversary gain to be equal to publisher loss. By varying this single parameter between \$0 and \$2,000 we were able to investigate a wide variety of scenarios, in which either the data publisher or the adversary is at an advantage.

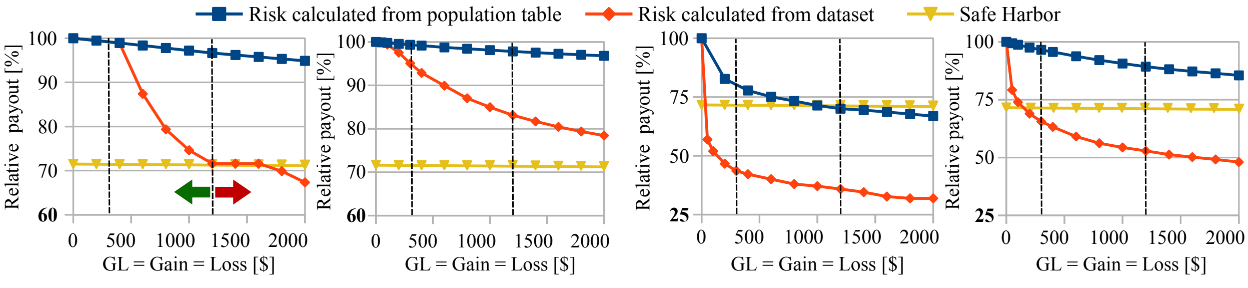
Performance Measures. When reporting results, we express the overall publisher payout relative to the theoretical maximum, which can be obtained when the data will never be attacked. In this case, the publisher payout equals (number of records) * (publisher benefit). This results in a maximal payout of 32,561 * \$1,200 = \$39.0732 million for our setup. We compared our approach to prior work and performed a runtime analysis of our implementation.

Results

Comparison with Prior Work

Wan et al. solved the game with record-level generalization using journalist risks calculated with a population table[16]. As mentioned, we developed methods to solve the game in the context record-level generalization and full-domain generalization with and without supplying a population table. Here, we present results of a comparison.

As a baseline, we used HIPAA Safe Harbor, which we applied by replacing all values of the attribute ZIP code with their initial 3 digits, masking zip codes with populations of less than 20,000 individuals and top-coding ages to 90 and above. The results of the experiment are shown in Figure 5. The first vertical line in each plot denotes the parameterization $GL = Gain = Loss = \$300$, which has been shown to be a realistic scenario[16]. The second vertical line indicates the point at which the data sharing scenario becomes disadvantageous for the publisher.



(a) Basic game: full-domain generalization + suppression

(b) Basic game: record-level generalization

(c) No-Attack variant: full-domain generalization + suppression

(d) No-Attack variant: record-level generalization

Figure 5: Publisher payout for the output of different implementations calculated using a population table.

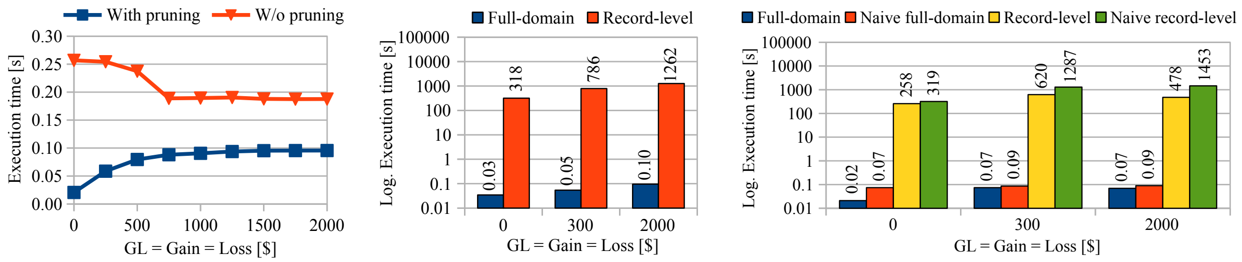
The results obtained for the basic game are shown in Figure 5 (a) and (b). There are a several notable findings to highlight. First, all of the methods provided a higher payout than Safe Harbor when the publisher was at an advantage. Second, it can be seen that, up to $GL = \$400$, full-domain generalization with risks calculated from the dataset resulted in a publisher payout that is close to that achieved using record-level generalization and a population table. This noteable because for GL = \$300 the method by Wan et al. achieved ony 0.17% more relative payout. Third, it can be seen that, when using a population table, full-domain generalization generally resulted in a payout similar to that obtained using record-level generalization. Even for GL = \$2000, the method by Wan et al. provided only 1.9% more relative payout. Finally, it can be observed that when calculating risks from the dataset, full-domain generalization resulted in a greater payout than record-level generalization up to about GL = \$600. This is because record-level generalization is much more flexible and, therefore, leads to a pronounced overgeneralization when risks are overestimated significantly.

Figures 5 (c) and (d) present the results obtained for the *No-Attack* variant of the game. It can be seen that this model can not be implemented effectively without supplying a population table. This is an intuitive finding as we have shown that the model implies a significant risk threshold (i.e., up to $\frac{1}{500} = 0.2\%$ for GL = \$2000). Still, it is notable that using full-domain generalization and a population table resulted in higher payout than Safe Harbor for $GL \leq \$800$.

Analysis of Scalability

We further investigated the scalability of our implementation in ARX. To do so, we (1) measured the impact of the pruning strategy, (2) compared execution times for full-domain generalization with record-level generalization and (3) measured the speed up achieved by implementing the *No-Attack* variant with k-anonymity or k-map. The experiments were performed on an Intel Core i5 3.1 GHz machine running a 64-bit Linux 3.2.0 kernel and a 64-bit JVM (1.7.0). Our method for record-level generalization using a population table is not included in the current release of ARX. Using full-domain generalization and a population table took around 2 seconds in the experiments described above. The results shown in Figure 6 focus on methods that calculate risks from the dataset.

Figure 6 (a) depicts the impact of the pruning strategy. It can be seen that the optimization was more effective in scenarios that were advantageous to the publisher. For example, we achieved a 4x speedup for GL = \$300 and only a



(a) Impact of the pruning strategy

(b) Methods for solving the game

(c) Methods for solving the *No-Attack* variant of the game

Figure 6: Execution times for solving the game without using a population table.

2x speedup for *GL* = $2,000. The main reason is that, when the scenario is less advantageous to the publisher, more generalization is required to obtain a good solution to the de-identification problem. This makes the optimization less effective and, consequently, increases execution times.

A comparison of execution times measured for full-domain generalization and for record-level generalization is shown in Figure 6 (b). It can be observed that full-domain generalization is four orders of magnitude more efficient than record-level generalization. This is reasonable, as the latter was implemented by executing full-domain generalization for every record in the dataset (i.e., 32,561 times).

Finally, Figure 6 (c) summarizes a comparison of the execution times measured for a naïve implementation of the *No-Attack* variant of the game and the optimized implementation proposed previously. It can be seen that implementing the model using traditional privacy models and de-identification algorithms reduced execution times by up to 67%. We measured more pronounced improvements for record-level generalization, which is significantly more computationally complex than full-domain generalization.

Discussion and Conclusions

In this paper, we introduced new variations of the game theoretic approach to health data de-identification and described their integration into the open source software ARX. In contrast to many other methods of data de-identification, the game theoretic approach explicitly accounts for the incentives of data publishers and recipients[16]. The experiments showed that our implementation provides a novel means for balancing publisher payout with execution times by using different models for transforming data. Our approach does not rely upon the existance of a population table, but users are free to specify one when it is available.

There are a wide variety of directions for further improving our implementation. In particular we believe our approach can be extended to support higher-dimensional data, such as genomic summary statistics, akin to the approach recently proposed for the minor allele frequencies of single nucleotide polymorphisms (SNPs)[17].

This is a natural extension because ARX is based on a modular design with a focus on scalability. The tool is already able to load very large and high-dimensional data, but two additional modules will need be developed to fully support the processing of such data. First, the existing implementation of ARX's generalization lattice[26] needs to be complemented with an implementation that represents attribute suppression policies as bit vectors. Second, the search algorithms currently implemented by the tool[25] need to be complemented with a genetic algorithm developed by Wan et al.[17]. This algorithm is generic in design (that is, it is not specific unto genomic data) and focuses solely on attribute suppression (i.e., either an attribute is retained or redacted). Since ARX supports attribute-level suppression, integration into the tool should be a feasible endeavour.

Acknowledgements

This work was sponsored, in part, from grants R01HG006844, R01LM009989, RM1HG009034, U01HG006385, and U01HG006378 from the National Institutes of Health.

References

[1] Denny JC, Bastarache L, Ritchie M, et al. Systematic comparison of phenome-wide association study of electronic medical record data and genome-wide association study data. Nat Biotechnol. 2013;31(12):1102–1111.

[2] Schneeweiss S. Learning from big health care data. N Engl J Med. 2014;370(23):2161–2163.

[3] Shameer K, Badgeley MA, Miotto R, Glicksberg BS, Morgan JW, Dudley JT. Translational bioinformatics in the era of real-time biomedical, health care and wellness data streams. Brief Bioinform. 2016;p. bbv118.

[4] US National Institutes of Health. Final NIH Statement on Sharing Research Data. Notice NOT-OD-03-032; 2003.

[5] National Science Foundation. Dissemination and sharing of research results; [Internet]. [cited 2017 Mar 04] Available from: `https://www.nsf.gov/bfa/dias/policy/dmp.jsp`.

[6] European Medicines Agency. EMA/240810/2013 – European Medicines Agency Policy on Publication of Clinical Data for Medicinal Products for Human Use; 2014.

[7] Liu V, Musen MA, Chou T. Data breaches of protected health information in the United States. J Am Med Assoc. 2015;313(14):1471–1473.

[8] Jensen PB, Jensen LJ, Brunak S. Mining electronic health records: towards better research applications and clinical care. Nat Rev Genet. 2012;13(6):395–405.

[9] Malin BA, Karp D, Scheuermann RH. Technical and policy approaches to balancing patient privacy and data sharing in clinical and translational research. J Investig Med. 2010;58(1):11–18.

[10] US Department of Health and Human Services. Standards for privacy of individually identifiable health information, Final Rule. 45 CFR, Parts 160–164. Federal Register. 2002;67(157):53182–53273.

[11] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Off J European Union. 2016 May;L119/59.

[12] El Emam K, Jonker E, Arbuckle L, Malin BA. A systematic review of re-identification attacks on health data. PloS One. 2011;6(12):e28071.

[13] Narayanan A, Felten EW. No silver bullet: de-identification still doesn't work. White Paper, Princeton U; 2014.

[14] Barth-Jones DC, El Emam K, Bambauer J, Cavoukian A, Malin BA. Assessing data intrusion threats. Science. 2015;348:6231.

[15] Duong Q, LeFevre K, Wellman MP. Strategic modeling of information sharing among data privacy attackers. Informatica. 2010;34:151–158.

[16] Wan Z, Vorobeychik Y, Xia W, et al. A game theoretic framework for analyzing re-identification risk. PloS One. 2015;10(3):e0120592.

[17] Wan Z, Vorobeychik Y, Xia W, Clayton EW, Kantarcioglu M, Malin BA. Expanding access to large-scale genomic data While promoting privacy: a game theoretic approach. Am J Hum Genet. 2017;100(2):316–322.

[18] Prasser F, Kohlmayer F. Putting statistical disclosure control into practice: the ARX data anonymization tool. In: Medical Data Privacy Handbook. New York: Springer; 2015. p. 111–148.

[19] European Union Agency for Network and Information Security (ENISA). Privacy and data protection by pesign – from policy to engineering. 2014;p. 1–79.

[20] Privacy Analytics. Privacy Analytics Eclipse; [Internet]. [cited 2017 Mar 04] Available from: `https://www.privacy-analytics.com/software/privacy-analytics-eclipse/`.

[21] European Medicines Agency. EMA/90915/2016 – External guidance on the implementation of the European Medicines Agency Policy on Publication of Clinical Data for Medicinal Products for Human Use; 2016.

[22] Dwork C. Differential privacy. In: Proc ICALP. Springer LNCS; 2006. p. 1–12.

[23] Samarati P, Sweeney L. Generalizing data to provide anonymity when disclosing information. In: Proc Symp Principles Database Systems; 1998. p. 188.

[24] El Emam K, Dankar FK. Protecting privacy using k-anonymity. J Am Med Inform Assoc. 2008;15(5):627–637.

[25] Prasser F, Bild R, Eicher J, Spengler H, Kohlmayer F, Kuhn KA. Lightning: utility-driven anonymization of high-himensional data. Trans Data Priv. 2016;9(2):161–185.

[26] Prasser F, Kohlmayer F, Kuhn KA. Efficient and effective pruning strategies for health data de-identification. BMC Med Inform Decis Mak. 2016;16(1):1.