Customized Recommendation System for Optimum Privacy Model Adoption

KONSTANTINOS PSARAFTIS[1], THEODOROS ANAGNOSTOPOULOS[1,2], KLIMIS NTALIANIS[1] AND NIKOLAOS MASTORAKIS[3]
[1]Department of Business Administration,
University of West Attica,
Agiou Spiridonos 28, Egaleo 12243,
GREECE
kostaspsaraftis@hotmail.com, kntal@teiath.gr
[2]Department of Infocommunication Technologies, ITMO University,
49 Kronverksky Pr., St. Petersburg 197101,
RUSSIA
thanag@teiath.gr
[3]Industrial Engineering Department
Technical University of Sofia,
Sofia,
BULGARIA
mastor@tu-sofia.bg

Abstract: - A large number of companies, organizations and other entities collect and elaborate personal data from people, which are frequently published for research or other promotional purposes. This paper deals with the effective anonymization, in applications that store data in relational databases. The optimum choice of a privacy model along with its application can the effectively protect personal data and allow low percentage of information loss. In this context and, as far as the need of anonymization of a relational database is concerned, a tool, named CRPM, has been developed, which is able to propose a suitable privacy model after the data recipient has viewed and answered questions related to the dataset to protect the disclosure of sensitive data.

Key-Words: – Anonymization, Privacy Model, Data Privacy, Relational Databases

1 Introduction

We are currently experiencing the information age where the collection of digital data from data holders such as governments, organizations or other entities has created great possibilities for the humanity to evolve. Gaining access to these data is of utmost importance because it can provide qualitative conclusions by data mining them. However, these datasets usually contain personal information. It is therefore the legal and ethical responsibility of the data holder to protect individual's anonymity by carefully applying a privacy algorithm to the dataset before its release to the public.

A privacy model expresses the privacy requirements a dataset should have. It also proposes an anonymization algorithm, which utilizes several techniques to fulfill those requirements. On the one hand excessive anonymization will result in distortions to the data, thus reducing its quality making it unsuitable for analysis and extraction of conclusions. On the other hand, poor anonymization will leave individuals unprotected against attackers. The goal is to anonymize the dataset in a way to ensure an individual's privacy and at the same time retain the data quality.

We identify that the first step to a successful anonymization is finding the most suitable

privacy model. To find it we need to specify the privacy requirements as precise as possible and consider the user's needs. Therefore, we have developed a tool, which asks, as an input from the user, the metadata of the dataset and as an output returns the most suitable privacy model.
The tool helps knowledgeable or not data holders and researchers find the precise privacy model their dataset need. Specifically, data holders with poor knowledge for anonymization could help them find with better chances and in a very short amount of time a suitable privacy model. Knowledgeable data holders who already selected a privacy model but wish to get a second opinion can cross check their decision. If the tool returns different results, it is a sign, they need to investigate the matter further. Lastly, researchers could enrich their knowledge on data privacy in relational databases in one place because the tool also provides detailed information on all related privacy models.

The rest of the paper is organized as follows. In Section II, we give a general description on anonymity and what has been proposed over the prior work privacy models, the tool supports. In Section III, we present the architecture of the proposed open-source privacy model recommendation tool. In Sections IV, we introduce the planning, the implementation and the results of the conducted experiments. Finally, Section V contains conclusions and future work.

2 Prior Work

There is an enormous literature on privacy in databases. We briefly mention the privacy models, the tool supports, starting with definition of anonymization.

Anonymization [4] refers to the privacy preserving data publishing approach that aims to de-associate the identity and sensitive data of record owners, as sensitive data must be kept for data analysis. In a relational database it is the process of transforming a table in the form of:

D(Explicit Identifier, Quasi Identifier, Sensitive Attributes, Non-Sensitive Attributes) to:
T (QID', Sensitive Attributes, Non-Sensitive Attributes)

where explicit identifiers are a set of attributes such as the social security number that contain information able to uniquely identify an individual. Quasi identifiers consist of attributes that could potentially link record owners to a specific data entry when combined. For instance, Sweeney [1] proved that the combination of Date of Birth, Sex and Postal Code is capable to identify 87% of individuals in the United States. Sensitive attributes consist of sensitive information such as disease while non-sensitive attributes contain the rest. For the anonymous table we remove explicit identifiers and keep the non-sensitive attributes. QID' is the result of applying anonymizing techniques to the QID attributes so that multiple records become indistinguishable with respect to QID'. Other options of making the original table D anonymous would be anatomization [5] which de-associates the relationship between quasi-identifier and sensitive attribute without editing the QID and random perturbation [6] which replaces original data with synthetic.

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

Table 1 – Original Table

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	
1	130**	<30	*	Heart Disease
2	130**	<30	*	Heart Disease
3	130**	<30	*	Viral Infection
4	130**	<30	*	Viral Infection
5	1485*	≥40	*	Cancer
6	1485*	≥40	*	Heart Disease
7	1485*	≥40	*	Viral Infection
8	1485*	≥40	*	Viral Infection
9	130**	3∗	*	Cancer
10	130**	3∗	*	Cancer

11	130**	3∗	*	Cancer
12	130**	3∗	*	Cancer

Table 2 – Anonymized Table

We categorize the privacy models based on the privacy threats to the dataset. There are three main privacy threats. The first, identity disclosure [1] (i.e., or record linkage), considers that a privacy threat happens when the adversary manages to make a one to one association between a record owner and a record in the published dataset. The second privacy threat is attribute disclosure [2] (i.e., or attribute linkage). Here the adversary may not precisely associate the identity of the target. Instead he/she could deduce the record's sensitive values of the record from out of the published data based on the sensitive values associated to the group that the target victim belongs. A grouping of rows based on the attributes Zip Code, Age, Nationality can be examined in Table 2. These privacy threats assume that the attacker knows in advance that the individual's information of the users is in the dataset. On the other hand, the last privacy threat, membership disclosure [3] (i.e., or table linkage), is the act of correlating the data owner with the published table itself. The presence or the absence of an individual in the dataset can reveal sensitive information. For example, identifying the presence of the target victim's record in a table concerning patients with cancer has already revealed sensitive information.

Table 3 shows the privacy models we will discuss, and their categorization based on their protection against privacy threats.

Privacy Model	***Privacy Threat***		
	Identity disclosure	Attribute disclosure	Membership disclosure
k-Anonymity	✓		
MultiR k-Anonymity	✓		
(c, t)-Isolation	✓		
l-Diversity	✓	✓	
Confidence Bounding		✓	
(X, Y) - Privacy	✓	✓	
(α, k)-Anonymity	✓	✓	
LKC- Privacy	✓	✓	
(k, e)-Anonymity		✓	
(ε, m)-Anonymity		✓	
t-Closeness		✓	
Personalized Privacy		✓	
FF- Anonymity		✓	
m-invariance	✓	✓	
δ-Presence			✓
ε-Differential Privacy			✓

Table 3 – Supported Privacy models.

k-anonymity [1] protects against identity disclosure attacks by ensuring that every record in the table with some qid values is indistinguishable from at least k − 1 other records with respect to QID. Table 2 shows a 4-anonymous table.

MultiRelational k-anonymity [8] studies privacy of personal data in multiple relational tables. Their model makes sure that each record owner will have at least k-1 other record owners with the same QID in the join of all the tables they appear.

The model *(c, t) isolation* [9] was developed to protects attacks against a statistical database. They propose that, for an adversary, every point in the dataset should be indistinguishable from at least t−1 other points. Point is considered the position of a record owner in the database and t is a threshold chosen according to social considerations. It is most effective if the statistical database has numerical attributes because it considers distances among data records.

To prevent attribute linkage attacks Machanavajjhala [2] proposes the *l-diversity* privacy model. They showed that an attacker can discover the values of sensitive attributes when there is little or no diversity. In Table 2 the fourth group of QID in the table has the same sensitive attribute. Therefore, privacy is breached if an attacker can associate the target victim with that group. The model requires that every qid group must contain at least l distinct sensitive values. In addition, l-diversity satisfies k-anonymity, where k=l, because each qid group contains at least l records.

Ke Wang [11] proposed *confidence bounding,* a privacy model which specifies privacy templates and limits the confidence of

inferring sensitive properties against threats caused by data mining abilities.

These privacy templates specify the sensitive property to be protected, the attributes identifying a group of individuals, and a maximum threshold for the confidence of inferring the sensitive property given the identifying attributes.

The above privacy models assume that each record represents a distinct individual. However, there are cases that several records could represent the same data owner. Therefore, each group of k records in a k-anonymous table would represent fewer than k data owners. To overcome this problem Ke Wang [7] proposed *(X-Y) Privacy* which specifies that each value on X will be linked to at least k distinct values on Y.

Wong [4] proposed *(a, k)-anonymity* model to protect both identity and attribute disclosure. The privacy model requires every qid in a table T to be shared by at least k records and conf(qid → s) ≤ a for any sensitive value s, where conf(qid → s) denotes the percentage of records containing s in the qid group. The rest, k and a are thresholds specified by the data holder.

In real life privacy attacks, it is very difficult for an adversary to acquire all the information in QID of the target. Based on that intuition Mohammed [12] proposes *LKC-privacy*, where L indicates the maximum values of the QID attributes the adversary knows about the target victim. LKC ensures that every combination of values in QIDj ⊆ QID with maximum length L in the data table T is shared by at least K records, and the confidence of inferring any sensitive values in S is not greater than C.

Most of the above privacy models assume the dataset contains categorical sensitive attributes. Zhang [13] proposes *(k, e)-anonymity* to address numerical sensitive attributes such as the salary. The privacy model splits the dataset into groups making sure that each group will contain at least k different sensitive values with a range of at least e.

The previous model ignores the classification of sensitive vales within a range λ. The problem is that if some sensitive values occur frequently within a subrange of λ, then the adversary can infer the subrange in a group. Jiexing [14] defines this type of attribute linkage attack as proximity attack. To protect a dataset from this *(ε, m)-anonymity* is introduced which demands that given a QI-group G, for every sensitive value x in G, at most 1/m of the tuples in G can have sensitive values "similar" to x, where the similarity is controlled by the threshold ε.

Li [15] observed that when the overall distribution of sensitive attributes is uneven, l-diversity is not effective against attribute linkage attacks. He proposed *t-closeness*, a privacy model which requires that the distribution of a sensitive attribute in any equivalence class must be close to the distribution of the attribute in the overall table. t-closeness uses the Earth Mover Distance (EMD) function to measure the closeness between two distributions of sensitive values, and requires the closeness to be within t.

Xiao and Yu [16] presented *Personalized Privacy,* to allow each record owner to specify his/her own privacy level. The previous models focus on a universal approach. As a result, the anonymized dataset could be offering insufficient protection to a subset of people, while applying excessive anonymization to another subset. Each sensitive attribute has a taxonomy tree where each record owner specifies his guarding node in the tree to perform the minimum generalization for satisfying everybody's requirements.

Most privacy models assume that the original table can be divided into QID and sensitive attributes. However, this assumption is not correct when an attribute contains both sensitive values and quasi-identifying values. Based on the above assumptions Wang [17] proposed *FF-anonymity* privacy model which models identifying/sensitive information at the value level, instead of at the attribute level.

Xiao and Tao [18] study the privacy issues in the dynamic data republishing model, in which they support re-publication of the microdata, after it has been updated with insertions and deletions. They developed a new generalization principle *m-invariance* that effectively limits the risk of privacy disclosure in re-publication.

To prevent membership disclosure, Ercan Nergiz [3] proposed the *δ-presence* privacy model. The general idea behind it is that the probability of inferring the presence of any potential target victim's record should be bound within a specified range δ = (δmin, δmax). δ is actually the primary value of anonymization and can has the property of to be interpreted in terms of increased risk of disclosure. This enables the

connection between the human-understandable policy and mathematical sound standards for anonymity.

Dwork [19] proposed *ε-differential privacy* based on the intuition that nothing about an individual should be learnable from a statistical database that cannot be learned without access to the database. The privacy model ensures data owners that they may submit their personal information to the database securely knowing that almost nothing can be discovered from the database with their information that could not be discovered without their information in it. It also ensures that attackers with random background knowledge have no power against the anonymized dataset.

3 Proposed System

The tool, CRPM (Customized Recommendation System for Optimum Privacy Model Adoption) is an ASP.Net MVC web application. It was developed using Microsoft Visual Studio Community 2017 with C# as the main programming language. In the next paragraph we will analyze its architecture and inner structure.

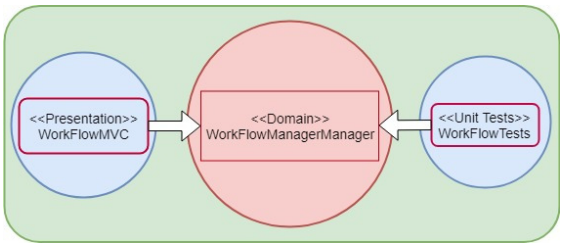

Fig 1 - CRPM Architecture

CRPM, follows all modern principles and is highly extendible for any researcher who is interested in effective data anonymity. As a solution, CRPM, is partitioned to different projects with respect to the Layered pattern. WorkFlowManager is the Domain layer that incorporates both behavior and data. It is an abstract layer and is agnostic to the upper layers as shown in figure 1. To continue, WorkFlowMVC is the presentation layer. We decided to apply the MVC (Model-View-Controller) pattern due to its wide usability in web applications, better separation of concerns and in order to decouple the domain with the presented entities. WorkFlowTests is the last project in our solution and contains unit tests. Unit tests are highly important not only to the tool itself but also to the research community. They provide documentation to the system, make the process agile and improve the quality of the code.

CRPM, ultimately, is an implementation of a decision tree of carefully selected questions. These questions were crafted to eliminate step by step privacy models. Figure 2 represents the workflow of the questions. Round shapes represent the questions and square the answers. We discuss with better detail the figure 2, in the following paragraph with the questions and answers its bullet represent.

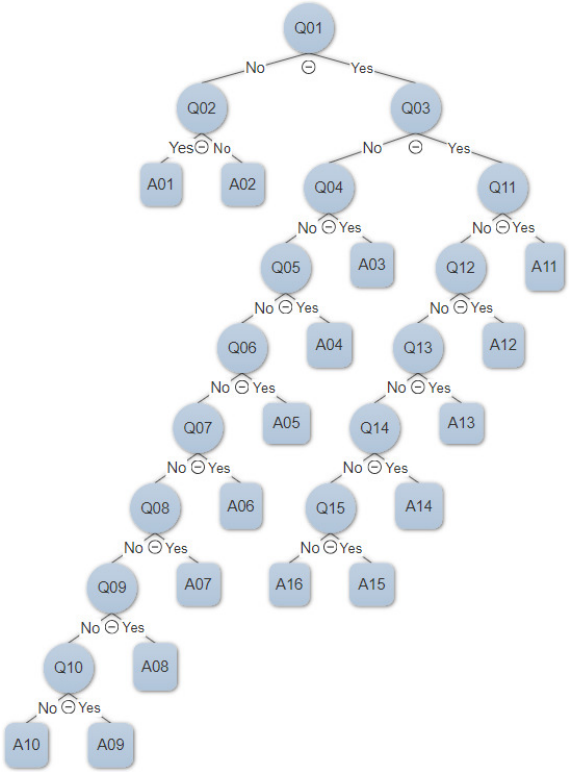

Fig. 2 - WorkFlow Questions

1. *Q01*: Does the adversary already know that the target victim exists in the database?
   From the feedback of the user we can now know if the privacy model belongs to the membership disclosure attack model or not.
2. *Q02*: Is the relational database statistic or not?
   If yes, then the tool suggests ε-differential privacy *(A01)*, else the tool returns δ-presence *(A02)*.

3. *Q03*: Do the sensitive attributes contain a rich variety of entries?
   The purpose of the question is to determine if the privacy model belongs to identity disclosure, if the answer is negative, or attribute disclosure if the answer is positive. Questions 04 to 10 are about identity disclosure, whereas 11 to 15 are about attribute disclosure.
4. *Q04*: Do more than one row entries belong to a single individual?
   If the answer is positive, the most acceptable privacy model is (X -Y) Privacy *(A03)*.
5. *Q05*: Does the dataset contain multiple tables?
   If yes, then the tool suggests MultiRelational k-Anonymity *(A04)*.
6. *Q06*: Are the contents of the dataset frequently updated?
   If yes, then the proper model is m-invariance *(A05)*.
7. *Q07*: Do the selected quasi-identifiers contain sensitive attributes?
   If yes, the most suitable model is FF-Anonymity *(A06)*.
8. *Q08*: Are you certain that the quasi-identifiers are precisely known?
   If yes, the tool returns k-anonymity *(A07)*.
9. *Q09*: Do most of the attributes contain numerical values?
   If yes, (c, t)-isolation is returned *(A08)*.
10. *Q10*: Are the sensitive attributes only numeric?
    If the answer is positive then (e, m)-anonymity *(A09)* is returned else (k, e)-anonymity *(A10)* is returned.
11. *Q11*: Is the dataset's dimensions extremely large?
    If yes, the tool suggests LKC-privacy *(A11)*.
12. *Q12*: Are you able to contact each record owner to ask for his privacy level demands?
    If yes, then the Personalized Privacy Preservation model *(A12)* is suitable.
13. *Q13*: Do you want to be able to specify protection level against specific sensitive attributes?
    If yes, then the suitable model is Confidence Bounding *(A13)*.
14. *Q14*: Do you wish to be able to determine qid-grouping size and the breach probability of each sensitive attribute?
    If yes, then (a, k)-anonymity *(A14)* is returned.
15. *Q15*: Are the sensitive attributes skewed distributed?
    Lastly, if the users respond positively, t-closeness *(A15) will* be returned, otherwise l-diversity *(A16)*.

In the lifetime of CRPM's session there is only one instance of the decision tree (named decisionTree). Using the singleton pattern, we instantiate this variable on the Global.asax.cs file which is one of the first classes to be executed in an ASP.NET MVC application. The code block is depicted in figure 3:

```
1. var decisionTree = new WizardManager.WizardSeed();
2. decisionTree.Initialize();
```

Fig. 3 - Singleton Code Block

Inside the WizardManager class, we can extend the tree structure with more questions, answers and information in privacy models.

So far, we have examined the architecture and design of CRPM. Following up we will analyze a use case as a specification tool for specifying the functional requirements of the software system. Suppose a data holder wishes to publish the anonymized version of a relational table concerning cancer survival statistics in his local area, the last 5 years. For this use case the examined sample was selected from random hospitals. Therefore, attackers cannot possibly know whether the possible target victim was part of the statistical data entries. These metadata and only are sufficient for the tool to present a suitable privacy model. Starting with question Q01 the data holder will respond negatively and the tool will direct him to the top left subtree of figure 2, which is Q02. Q02 will ask the user if the table is statistic or not and the user will respond positively. CRPM will finally suggest the data holder that the optimum privacy model to adopt is ε-differential privacy *(A01)*. Note that these answers constitute the path to the tree as shown in figure 2. Lastly, the resulting page will provide the user with external Uniform Resource

Locator containing further information regarding the privacy model.

4 Experimental Results

4.1 Setup

For the experimental setup of CRPM we attended the National Technical University of Athens (NTUA). 30 out of 125 undergraduate students who were related to information security courses decided to participate in our survey. The experimental setup is separated to three phases, namely *PHASE I, II,* and *III* respectively.

Starting with *PHASE I*, we had the students find a public relational database. They carefully examined it and selected the privacy model they found most suitable. We also accommodated for those who did not have previous knowledge on data privacy and provided them with all the necessary information. For *PHASE II* we granted the students access to CRPM. We asked them to carefully go through the wizard's questions and finally store the outcome. Finally, for *PHASE III* we prepared a questionnaire and asked the students to go through it. We tried to make sure, the survey's questions were specific, objective and understandable. Our goal was to get a feedback from the students and uncover answers relative to CRPM.

4.2 Results

We present the questions and answers according to the proposed experiment.

1. How helpful was the information page included in CRPM?

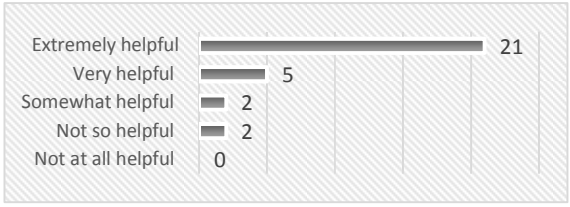

Fig. 4 – Answers to Question 1

- 21 students (70%) chose "Extremely helpful"
- 5 students (16%) chose "Very helpful"
- 2 students (7%) chose "Somewhat helpful"
- 2 students (7%) chose "Not so helpful"
- No students (0%) chose "Not at all helpful"

2. Overall, how easy to use do you find using CRPM?

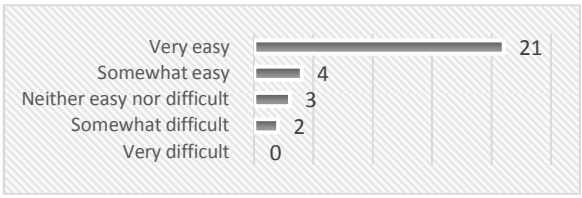

Fig. 5 – Answers to Question 2

- 21 students (70%) chose "Very easy"
- 4 students (13%) chose "Somewhat easy"
- 3 students (10%) chose "Neither easy nor difficult"
- 2 students (7%) chose "Somewhat difficult"
- No students (0%) chose "Very difficult"

3. How close are your results over the database against CRPM's results?

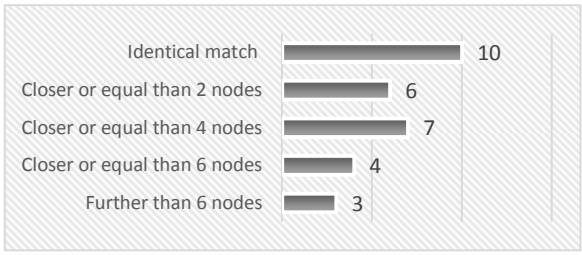

Fig. 6 – Answers to Question 3

- 10 students (33%) chose "Identical match"
- 6 students (20%) chose "Closer or equal than 2 nodes"
- 7 students (23%) chose "Closer or equal than 4 nodes"
- 4 students (14%) chose "Closer or equal than 6 nodes"
- 3 students (10%) chose "Further than 6 nodes"

4. Overall, how satisfied or dissatisfied are you with CRPM?

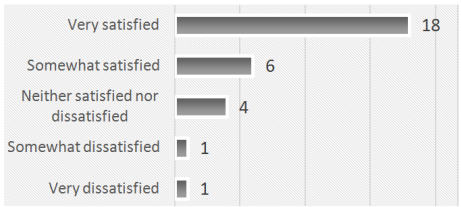

Fig. 7 – Answers to Question 4

- 18 students (60%) chose "Very satisfied"
- 6 students (20%) chose "Somewhat satisfied"
- 4 students (13%) chose "Neither satisfied nor dissatisfied"
- 1 student (3.5%) chose "Somewhat dissatisfied"
- 1 student (3.5%) chose "Very dissatisfied"

5. How likely would you recommend CRPM to another researcher?

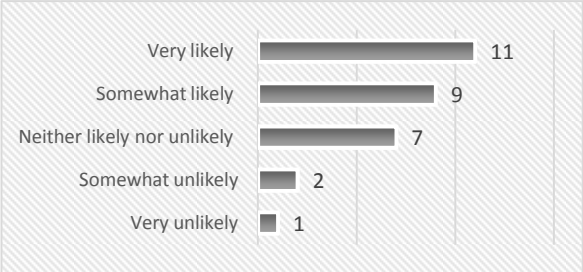

Fig. 8 – Answers to Question 5

- 11 students (37%) chose "Very likely"
- 9 students (30%) chose "Somewhat likely"
- 7 students (22.5%) chose "Neither likely nor unlikely"
- 2 students (7%) chose "Somewhat unlikely"
- 1 student (3.5%) chose "Very unlikely"

4.3 Discussion

The experimental procedure required 3 phases. Students followed the procedure literally and no problem occurred during the phases.

Starting with *Question 1,* we expected the positive results as CRPM contains a separate section with information on all supported privacy models and a rich variety of material in data privacy. In addition, *Question 2* was also expected. CRPM offers a simple and user-friendly interface. Students experienced no difficulty in the use of the tool. Furthermore, extracting metadata from a relational database is difficult and each researcher gives unique results. We tried to design a workflow as precise as possible given a set of metadata and privacy requirements. The positive feedback from *Question 3* is an affirmative indicator of our good work on the workflow. Lastly, for *Questions 4 and 5* the positive feedback is somewhat related to the results in the previous 3 questions. These depict the well performance of CRPM and trust, the research community has for the tool.

5 Conclusions and Future Work

Publishing collected data in such a way that unquestionably ensures total anonymity is still an open matter. Many privacy models have been proposed as far as relational databases are concerned. We have examined the most accepted of them and categorized them based on the privacy threat they protect. Identifying correctly the privacy requirements of the dataset and selecting a suitable anonymization algorithm insures data anonymity on the one hand and high data quality on the other hand. Consequently, we developed CRPM, a web tool that aims to propose the most suitable privacy model based on the dataset's attributes. CRPM is highly extensible. The main and most useful extension would be to include privacy models for all types of data storage. Lastly, it could import the dataset, besides user input, to infer useful information and export conclusions.

References:

[1] L. Sweeney. k-Anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570, 2002.

[2] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. "l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1), March 2007.

[3] M. Ercan Nergiz, M. Atzori, and C. W. Clifton. Hiding the presence of individuals from shared databases. *In Proc. of ACM International Conference on Management of Data (SIGMOD)*, pages 665–676, Vancouver, Canada, 2007.

[4] R. C. W. Wong, J. Li., A. W. C. Fu, and K. Wang. (α, k)-anonymity: An enhanced k-anonymity model for privacy preserving data publishing. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 2006.

[5] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. *In Proc. of the 32nd Very Large Data Bases (VLDB)*, Seoul, Korea, September 2006.

[6] X. Xiao, Y. Tao, and M. Chen. Optimal random perturbation at multiple privacy levels. *In Proc. of the 35th Very Large Data Bases (VLDB)*, pages 814–825, 2009.

[7] K. Wang and B. C. M. Fung. Anonymizing sequential releases. *In Proc. of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 414–423, Philadelphia, PA, August 2006.

[8] M. Ercan Nergiz, C. Clifton, and A. Erhan Nergiz. Multirelational k-anonymity. *In Proc. of the 23rd International Conference on Data Engineering (ICDE)*, pages 1417–1421, Istanbul, Turkey, 2007.

[9] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. *In Proc. of Theory of Cryptography Conference (TCC)*, pages 363–385, Cambridge, MA, February 2005.

[10] M. Terrovitis and N. Mamoulis. Privacy preservation in the publication of trajectories. *In Proc. of the 9th International Conference on Mobile Data Management (MDM)*, pages 65–72, April 2008.

[11] K. Wang, B. C. M. Fung, and P. S. Yu. Handicapping attacker's confidence: *An alternative to k-anonymization. Knowledge and Information Systems (KAIS)*, 11(3):345–368, April 2007

[12] N. Mohammed, B. C. M. Fung, P. C. K. Hung, and C. K. Lee. Anonymizing healthcare data: A case study on the blood transfusion service. *In Proc. of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1285–1294, Paris, France, June 2009.

[13] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate query answering on anonymized tables. *In Proc. of the 23rd IEEE International Conference on Data Engineering (ICDE)*, April 2007.

[14] J. Li, Y. Tao, and X. Xiao. Preservation of proximity privacy in publishing numerical sensitive data. *In Proc. of the ACM International Conference on Management of Data (SIGMOD)*, pages 437–486, Vancouver, Canada, June 2008.

[15] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and "l-diversity. *In Proc. of the 21st IEEE International Conference on Data Engineering (ICDE), Istanbul, Turkey, April 2007.*

[16] X. Xiao and Y. Tao. Personalized privacy preservation. *In Proc. of ACM International Conference on Management of Data (SIGMOD), Chicago, IL, 2006.*

[17] K. Wang, Y. Xu, A. W. C. Fu, and R. C. W. Wong. ff-anonymity: When quasi-identifiers are missing. *In Proc. of the 25th IEEE International Conference on Data Engineering (ICDE), March 2009.*

[18] X. Xiao and Y. Tao. m-invariance: Towards privacy preserving republication of dynamic datasets. *In Proc. of ACM International Conference on Management of Data (SIGMOD), Beijing, China, June 2007.*

[19] C. Dwork. Differential privacy: A survey of results. *In Proc. of the 5th International Conference on Theory and Applications of Models of Computation (TAMC)*, pages 1–19, Xian, China, April 2008.