Inference-Proof Data Publishing by Minimally Weakening a Database Instance

Joachim Biskup **Marcel Preuß**

Information Systems and Security (ISSI)

Technische Universität Dortmund, Germany

October 13, 2014

technische universität dortmund

Table of Contents

Context of this Work

Inference-Proof Data Publishing

Nowadays: Data publishing is ubiquitous

- Governments and companies provide data
- People share data about their private lifes

But: Original data often contains sensitive (personal) information

- Set up a confidentiality policy
- Release only "inference-proof views" of original data
  - No information to be protected is revealed
  - Even if an adversary tries to deduce inferences

Supposed Database Setting

Relational schema $\langle R|\mathcal{A}_R|\emptyset\rangle$

- Relational symbol R
- Attribute set $\mathcal{A}_R = \{A_1, \ldots, A_n\}$
- No database constraints declared (for now)
- Infinite set *Dom* of constant symbols

Complete relational instance r over $\langle R|\mathcal{A}_R|\emptyset\rangle$

- Finite number of valid database tuples over *Dom*
- CWA: Each constant combination not contained in r is invalid
  - Infinite number of invalid tuples
  - No constant combination is undefined

First-Order Logic for Modeling Databases

Given first-order language $\mathscr{L}$ with equality

- Predicate symbol R with arity $|\mathcal{A}_R| = n$
- Predicate symbol $\equiv$ for expressing equality
- Infinite set Dom of constant symbols

Database-specific semantics: $\mathcal{I}$ is DB-Interpretation, if

- Dom is the universe of $\mathcal{I}$ and $\mathcal{I}(v) = v$ for each $v \in Dom$,
- R interpreted by finite $\mathcal{I}(R) \subset Dom^n$,
- $\equiv$ interpreted by $\mathcal{I}(\equiv) = \{(v, v) \mid v \in Dom\}$

Logic-Oriented Modeling of Relational Instances

Given instance r:

$+$	$-$
(a, b, c)	(a, a, a)
(a, c, c)	(a, a, b)
(b, a, c)	(a, a, c)
	$\vdots$

$R(a, b, c),\ R(a, c, c),\ R(b, a, c)$

$$(\forall X)(\forall Y)(\forall Z)\ [\\ (X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee \\ (X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee \\ (X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee \\ \neg R(X, Y, Z) \quad]$$

Idea of logic-oriented modeling:

- Each valid tuple as corresponding ground atom
- Infinite set of invalid tuples as completeness-sentence
  - List all tuples which are not invalid ($\rightarrow$ Finite set)
  - All other tuples are invalid ($\rightarrow$ Infinitely many)

Confidentiality Policy

Confidentiality policy *psec*

- Finite set of potential secrets
- Potential secret: Ground atom $R(\boldsymbol{c})$ with $\boldsymbol{c} \in Dom^n$

Semantics of potential secret $\Psi \in psec$

- If Ψ is valid in r: Adversary **must not** get to know this
- Otherwise: Adversary may know that Ψ is invalid in r

Assume: Adversary is aware of policy

Inference-Proof Weakenings

Definition of Inference-Proofness

Given:

- Complete original instance r over $\langle R|\mathcal{A}_R|\emptyset\rangle$
- Confidentiality policy *psec*
- Weakening algorithm $weak(r, psec)$

Inference-Proofness: From adversary's point of view

- For each potential secret $\Psi \in psec$
- Existence of complete alternative instance r^Ψ over $\langle R|\mathcal{A}_R|\emptyset\rangle$
  - r^Ψ does **not** satisfy Ψ
  - r^Ψ is indistinguishable from original instance r
    $\rightarrow weak(r^\Psi, psec) = weak(r, psec)$

Case Study 1: Given Setting

Policy: $psec = \{\, \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,c,c) \,\}$

Original instance r:

+	−
(a,b,c)	(a,a,a)
(a,c,c)	(a,a,b)
(b,a,c)	(a,a,c)
	$\vdots$

$R(a,b,c),\ R(a,c,c),\ R(b,a,c)$

$$(\forall X)(\forall Y)(\forall Z)\ [\ (X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee (X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee (X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee \neg R(X,Y,Z)\ ]$$

Obviously: $\mathcal{I}_r \models_M \Psi_1,\ \mathcal{I}_r \models_M \Psi_2$

Case Study 1: Weakening

Policy: $psec = \{\, \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,c,c) \,\}$

Weakening $weak\,(r, psec)$:

+	−
~~(a, b, c)~~	(a, a, a)
~~(a, c, c)~~	(a, a, b)
(b, a, c)	(a, a, c)
	⋮

Disjunctive knowledge:
$R(a,b,c) \vee R(a,c,c)$

$R(b,a,c)$

$R(a,b,c) \vee R(a,c,c)$

$$(\forall X)(\forall Y)(\forall Z)\,[\ (X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee (X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee (X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee \neg R(X,Y,Z)\ ]$$

Achievement: $weak\,(r, psec) \not\models_{DB} \Psi_1,\ weak\,(r, psec) \not\models_{DB} \Psi_2$

Case Study 1: Alternative Instance Protecting Ψ_1

Policy: $psec = \{ \Psi_1 = R(a, b, c),\ \Psi_2 = R(a, c, c) \}$

Alternative instance r^{Ψ_1} from adversary's POV:

+	−
	(a, a, a)
(a, c, c)	(a, a, b)
(b, a, c)	$\vdots$
	(a, b, c)
	$\vdots$

Question: Is r^{Ψ_1} credible from adversary's POV?

Adversary's view: $\mathcal{I}_{r^{\Psi_1}} \not\models_M \Psi_1$, $\mathcal{I}_{r^{\Psi_1}} \models_M \Psi_2$

Case Study 1: Indistinguishability of Instance r^{Ψ_1}

Policy: $psec = \{\, \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,c,c) \,\}$

Adversary's simulation of $weak\,(r^{\Psi_1}, psec)$:

+	−
	(a,a,a)
~~(a,c,c)~~	(a,a,b)
(b,a,c)	$\vdots$
	~~(a,b,c)~~
	$\vdots$

Disjunctive knowledge:
$R(a,b,c) \vee R(a,c,c)$

$R(b,a,c)$

$R(a,b,c) \vee R(a,c,c)$

$$(\forall X)(\forall Y)(\forall Z)\ [\ (X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee (X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee (X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee \neg R(X,Y,Z)\ ]$$

r^{Ψ_1} and r are indistinguishable: $weak\,(r^{\Psi_1}, psec) = weak\,(r, psec)$

technische universität dortmund

Case Study 1: Alternative Instance Protecting Ψ_2

Policy: $psec = \{ \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,c,c) \}$

Alternative instance r^{Ψ_2} from adversary's POV:

+	−
(a,b,c)	(a,a,a)
	(a,a,b)
(b,a,c)	$\vdots$
	(a,c,c)
	$\vdots$

Question: Is r^{Ψ_2} credible from adversary's POV?

Again: Simulation of $weak(r^{\Psi_2}, psec)$

Adversary's view: $\mathcal{I}_{r^{\Psi_2}} \models_M \Psi_1$, $\mathcal{I}_{r^{\Psi_2}} \not\models_M \Psi_2$

technische universität dortmund

Case Study 2: Given Setting

Policy: $psec = \{ \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,b,d) \}$

Original instance r:

+	−
(a,b,c)	(a,a,a)
(a,c,c)	(a,a,b)
(b,a,c)	$\vdots$
	(a,b,d)
	$\vdots$

$R(a,b,c),\ R(a,c,c),\ R(b,a,c)$

$$
\begin{aligned}
&(\forall X)(\forall Y)(\forall Z)\ [\\
&(X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee\\
&(X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee\\
&(X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee\\
&\neg R(X,Y,Z)\ ]
\end{aligned}
$$

Obviously: $\mathcal{I}_r \models_M \Psi_1,\ \mathcal{I}_r \not\models_M \Psi_2$

Case Study 2: Weakening

Policy: $psec = \{\, \Psi_1 = R(a,b,c),\ \Psi_2 = R(a,b,d)\,\}$

Weakening $weak(r, psec)$:

+	−
~~(a, b, c)~~	(a, a, a)
(a, c, c)	(a, a, b)
(b, a, c)	⋮
	~~(a, b, d)~~
	⋮

Disjunctive knowledge:
$R(a,b,c) \vee R(a,b,d)$

$R(a,c,c),\ R(b,a,c)$

$R(a,b,c) \vee R(a,b,d)$

$$(\forall X)(\forall Y)(\forall Z)\ [\ (X \equiv a \wedge Y \equiv b \wedge Z \equiv c) \vee (X \equiv a \wedge Y \equiv b \wedge Z \equiv d) \vee (X \equiv a \wedge Y \equiv c \wedge Z \equiv c) \vee (X \equiv b \wedge Y \equiv a \wedge Z \equiv c) \vee \neg R(X,Y,Z)\ ]$$

Achievement: $weak(r, psec) \not\models_{DB} \Psi_1,\ weak(r, psec) \not\models_{DB} \Psi_2$

Case Study 3: The Easy Case

Policy: $psec = \{\, \Psi_1 = R(a,a,a),\ \Psi_2 = R(a,a,b)\,\}$

Original instance r:

+	−
(a,b,c)	(a,a,a)
(a,c,c)	(a,a,b)
(b,a,c)	(a,a,c)
	⋮

Nothing to weaken!

Neither Ψ_1 nor Ψ_2 need to be protected.

$\rightarrow weak(r, psec) := r$

Obviously: $\mathcal{I}_r \not\models_M \Psi_1$, $\mathcal{I}_r \not\models_M \Psi_2$

Clustering of Non-Simple Policies (1)

How to deal with non-simple policies of an arbitrary size?

- Partition the policy into a set of disjoint clusters
- For each cluster C: Construct disjunction $\bigvee_{\Psi \in C} \Psi$

How to achieve only meaningful disjunctions?

- Declare a set of admissible clusters
  $\rightarrow$ Employ high level languages such as SQL
- Goal: Each admissible disjunction should be well-balanced
  - Provide as much useful information as possible
  - All alternatives provided should be equally probable
- Only admissible clusters allowed in final disjoint clustering

Clustering of Non-Simple Policies (2)

How to balance availability and confidentiality requirements?

- Size of cluster C
  induces length of disjunction $\bigvee_{\Psi \in C} \Psi$
- Length of disjunction $\bigvee_{\Psi \in C} \Psi$
  induces number of alternative instances
  protecting a policy element of cluster C

In the following: Goal is to maximize availability

- Keep size of clusters as small as possible
- Only one alternative instance per potential secret required
  $\rightarrow$ Clusters of size 2 comply with security definition

Preparing the Clustering Algorithm

Requirements for clustering summarized

1. Each cluster is of size 2 (Maximizing availability)
2. Each cluster is admissible (Meaningful clusters)
3. Different clusters are pairwise disjoint } (Partitioning)
4. Each policy element is in a cluster } (Partitioning)

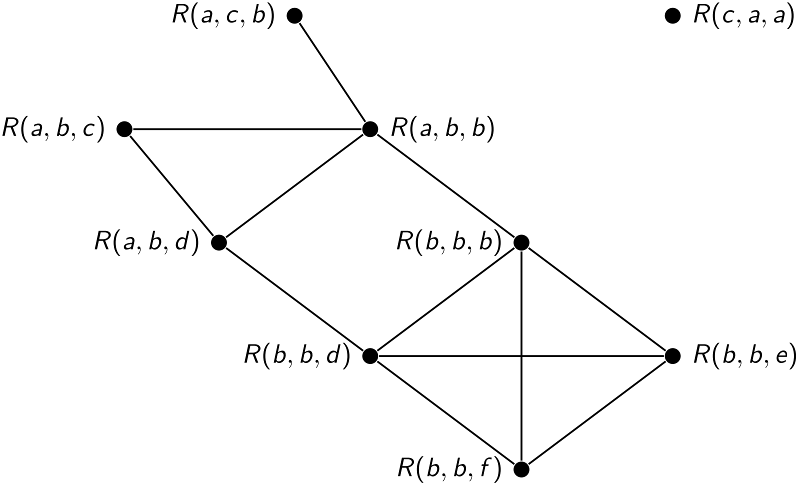
How to implement this **efficiently** on the **operational level**?

Model all admissible clusters within simple and undirected **Indistinguishability-Graph** $G = (V, E)$ with

- $V := psec$
- $E := \{\, \{\Psi_1, \Psi_2\} \in V \times V \mid \Psi_1 \vee \Psi_2 \text{ is admissible} \,\}$

technische universität dortmund

Example: Indistinguishability-Graph



First Idea for Clustering Algorithm

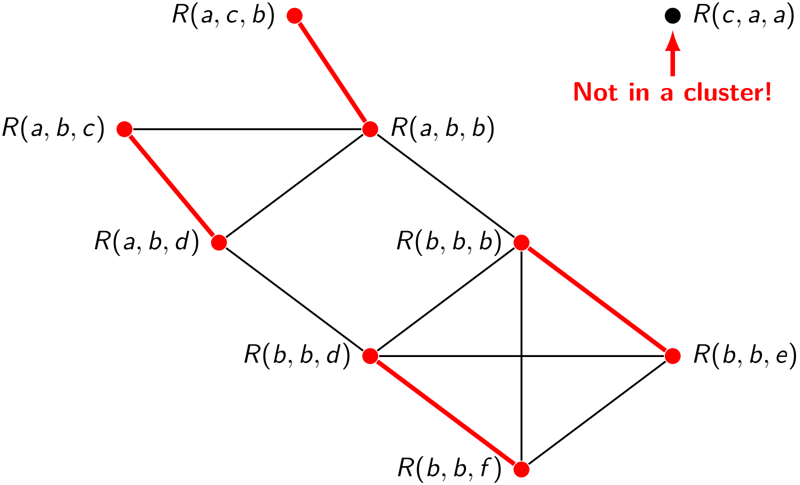
Compute maximum matching M on indistinguishability-graph G

- $M \subseteq E$ is a matching on G, if
  each pair of different $\{\Psi_1, \Psi_2\}, \{\bar{\Psi}_1, \bar{\Psi}_2\} \in M$ is disjoint
- M is maximum if there is no matching M' with $|M'| > |M|$

Is a maximum matching M on G the wanted clustering?

1. Each cluster is of size 2 ✓
2. Each cluster is admissible ✓
3. Different clusters are pairwise disjoint ✓
4. There may be policy elements not contained in a cluster ↯
   (Although matching is maximum)

Example: Clustering by Maximum Matching



Improved Idea for Clustering Algorithm

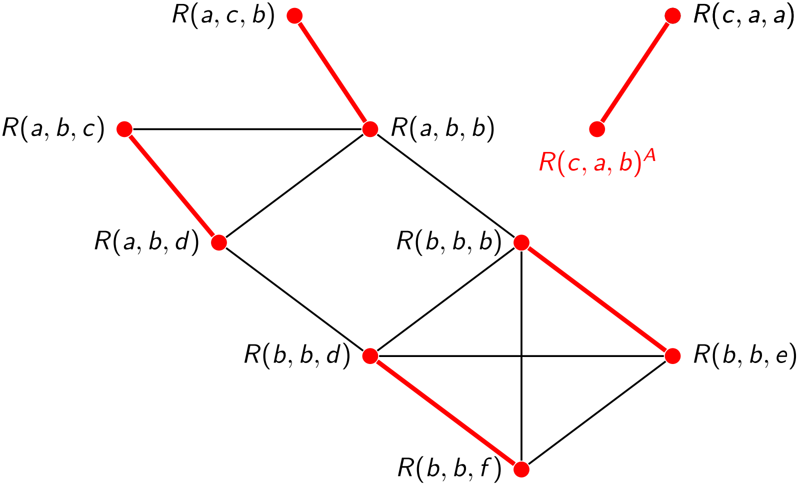
How to ensure that each policy element is in a cluster?

- Compute a maximum matching M
- Compute a matching extension M^* of M
  - Initially: $M^* := M$
  - For each potential secret Ψ not covered by M
    - Create a **suitable** additional potential secret Ψ^A for Ψ
    - Add cluster $\{\Psi, \Psi^A\}$ to M^*

How to create a **suitable** additional potential secret Ψ^A for Ψ?

- Create ground atom $\Psi^A = R(\boldsymbol{c})$
- Ensure that Ψ^A is not in the policy and not yet in M^*
- Ensure that $\Psi \vee \Psi^A$ would be admissible if Ψ^A was in policy

Example: Matching Extension



technische universität dortmund

Creation of Weakened Instance

Assume: Clustering M_r^* is given s.t. for each cluster $\{\Psi_1, \Psi_2\}$ the original instance r satisfies Ψ_1 or Ψ_2

Construction of weakened instance $weak(r, psec)$:

- Positive knowledge: Ground atom $R(\boldsymbol{c})$ for each $\boldsymbol{c} \in r$ with $R(\boldsymbol{c}) \not\models_{DB} \Psi$ for each $\Psi \in \bigcup_{C \in M_r^*} C$
- Disjunctive knowl.: Disjunction $\Psi_1 \vee \Psi_2$ for each cluster $\{\Psi_1, \Psi_2\} \in M_r^*$
- Negative knowledge: Each constant combination neither in positive knowledge nor in a disjunction is not valid by completeness sentence

The Overall Algorithmic Approach

Algorithm to compute weakenings

Inputs: original instance r, confidentiality policy $psec$

- **Stage 1:** Clustering of potential secrets (independent of r)
  - Generate indistinguishability-graph $G = (V, E)$ from $psec$
  - Compute maximum matching $M \subseteq E$ on G
  - Construct extended matching M^* based on M
- **Stage 2:** Creation of weakened instance (dependent on r)
  - Create set of clusters with a policy element not obeyed by r: $M_r^* := \{ \{\Psi_1, \Psi_2\} \in M^* \mid \mathcal{I}_r \models_M \Psi_1 \text{ or } \mathcal{I}_r \models_M \Psi_2 \}$
  - Create weakened instance $weak(r, psec)$ based on r and M_r^*

Example: Stage 2 of Weakening Algorithm

Clustering: $\{\ \{R(a,b,b),\ R(a,c,b)\},\ \{R(a,b,c),\ R(a,b,d)\}$
$\{R(b,b,b),\ R(b,b,e)\},\ \{R(b,b,d),\ R(b,b,f)\}$
$\{R(c,a,a),\ R(c,a,b)^{A}\}\ \}$

Instance r:

+	−
(a,b,a)	(a,a,a)
(a,b,b)	(a,a,b)
(a,c,b)	$\vdots$
(c,a,b)	

Instance $weak(r, psec)$:

$R(a,b,a)$

$R(a,b,b) \vee R(a,c,b)$
$R(c,a,a) \vee R(c,a,b)$

$$
\begin{aligned}
&(\forall X)(\forall Y)(\forall Z)\ [\\
&(X \equiv a \wedge Y \equiv b \wedge Z \equiv a) \vee\\
&(X \equiv a \wedge Y \equiv b \wedge Z \equiv b) \vee\\
&(X \equiv a \wedge Y \equiv c \wedge Z \equiv b) \vee\\
&(X \equiv c \wedge Y \equiv a \wedge Z \equiv a) \vee\\
&(X \equiv c \wedge Y \equiv a \wedge Z \equiv b) \vee\\
&\neg R(X,Y,Z)\ ]
\end{aligned}
$$

Inference-Proofness: Sketch of Proof (1)

Consider arbitrary $\tilde{\Psi} \in psec$

Suppose: $\tilde{\Psi}$ is in cluster $\{\tilde{\Psi}, \tilde{\Psi}_I\}$

Case 1: $\mathcal{I}_r \not\models_M \tilde{\Psi} \vee \tilde{\Psi}_I$

- Construct alternative instance $r^{\tilde{\Psi}} := r$
- $r^{\tilde{\Psi}}$ obeys $\tilde{\Psi}$: $\mathcal{I}_{r^{\tilde{\Psi}}} \not\models_M \tilde{\Psi} \vee \tilde{\Psi}_I$ implies $\mathcal{I}_{r^{\tilde{\Psi}}} \not\models_M \tilde{\Psi}$ ✓
- Indistinguishability: $r^{\tilde{\Psi}} = r$ by construction of $r^{\tilde{\Psi}}$
  $\rightarrow weak(r^{\tilde{\Psi}}, psec) = weak(r, psec)$ ✓

technische universität dortmund

Inference-Proofness: Sketch of Proof (2)

Case 2: $\mathcal{I}_r \models_M \tilde{\Psi} \vee \tilde{\Psi}_I$

- Construct alternative instance $r^{\tilde{\Psi}} := (r \setminus \{\tilde{\Psi}\}) \cup \{\tilde{\Psi}_I\}$
- $r^{\tilde{\Psi}}$ obeys $\tilde{\Psi}$: $\mathcal{I}_{r^{\tilde{\Psi}}} \not\models_M \tilde{\Psi}$ by construction of $r^{\tilde{\Psi}}$ ✓
- Indistinguishability:
  For each cluster $\{\Psi, \Psi_I\}$: $\mathcal{I}_{r^{\tilde{\Psi}}} \models_M \Psi \vee \Psi_I$ iff $\mathcal{I}_r \models_M \Psi \vee \Psi_I$
  - For cluster $\{\tilde{\Psi}, \tilde{\Psi}_I\}$: $\mathcal{I}_{r^{\tilde{\Psi}}} \models_M \tilde{\Psi} \vee \tilde{\Psi}_I$ by construction of $r^{\tilde{\Psi}}$
  - For each other $\{\Psi, \Psi_I\}$: $\mathcal{I}_{r^{\tilde{\Psi}}} \models_M \Psi \vee \Psi_I$ iff $\mathcal{I}_r \models_M \Psi \vee \Psi_I$
    by construction of $r^{\tilde{\Psi}}$ and by disjoint clusters

$\rightarrow$ $weak(r^{\tilde{\Psi}}, psec) = weak(r, psec)$ ✓

Experimental Evaluation of Approach

About the prototype implementation

- Sample indistinguishability criterion based on local distortion
- Graph constructed with a flavor of merge-join algorithm
- Boost-Library employed for maximum matching computation

Lessons learned from evaluation of prototype

- Algorithm can handle instances and policies of realistic size
- Runtime of Stage 2 is negligible
- Runtime of Stage 1 is dominated by matching computation
- Stage 1 is significantly faster with matching heuristic
  $\rightarrow$ Slight loss of availability ($\rightarrow$ more unmatched vertices)

Extending the Approach

technische universität dortmund

Existentially-Quantified Atoms as Potential Secrets

Now: Improve expressiveness of potential secrets

Existentially quantified atoms like $(\exists \boldsymbol{X})\, R(t_1, \ldots, t_n)$ in policy

- Each t_i is either a constant of *Dom* or a variable of $\boldsymbol{X}$
- Each variable is existentially quantified
- Each variable occurs only once in $t_1, \ldots, t_n$

New difficulty arising: Too strong formulas

- Consider: $R(a, b, c) \vee (\exists X)\, R(a, b, X)$
- Adversary must believe $R(a, b, c)$ to protect $(\exists X)\, R(a, b, X)$
- But: $R(a, b, c)$ directly implies $(\exists X)\, R(a, b, X)$ ↯

Cleaned Confidentiality Policy

Avoid too strong formulas by cleaning the policy

- Identify a maximum subset of logically weakest sentences (Without semantically equivalent sentences)
- Remove all other sentences from policy

Properties of cleaned confidentiality policy

- All alternatives provided by disjunctions are weakest sentences of policy $\rightarrow$ Do not imply other sentences of (original) policy
- Knowledge protected by removed stronger sentences is still protected by remaining weaker sentences

technische universität dortmund

A Basic Kind of A Priori Knowledge

Usually: Adversary also has some a priori knowledge

- Set of sentences *prior* (containing database constraints)
- Original instance *r* must satisfy *prior*
- *prior* must not imply a sentence of the confidentiality policy

New difficulty arising: Each alternative instance must also satisfy *prior* to be credible

So far: Inference-proofness under *prior* of ground atoms $R(\boldsymbol{c})$

- $R(\boldsymbol{c})$ satisfied by original instance
- $R(\boldsymbol{c})$ does not imply a $\Psi \in psec$

} $R(\boldsymbol{c})$ as **atom** in weakening

- Atoms of (positive part of) weakening in alternative instances

Conclusion & Future Work

technische universität dortmund

Conclusion & Future Work

Our contribution:

- Approach creating inference-proof materialized views
- Therefore: Replace some definite information by disjunctions
- Limited expressiveness $\rightarrow$ Efficient computation

Possible future work:

- Commonly used database constraints as a priori knowledge $\rightarrow$ Equality/Tuple Generating Dependencies
- Guarantee a certain number of $k > 2$ different “secure” alternative instances for each potential secret
- Elaborate connection to k-anonymity/ℓ-diversity