Linking Sensitive Data: Approaches and Vulnerabilities

Peter Christen and **Anushka Vidanage**

School of Computing and Software Innovation Institute
College of Engineering and Computer Science,
The Australian National University, Canberra

Contact: **peter.christen@anu.edu.au / anushka.vidanage@anu.edu.au**

Motivating example: A pandemic

Motivating example: Data science

- Understanding (or even preventing) the outbreak of a pandemic requires identifying unusual patterns of symptoms, ideally in real time
- Data from many different sources will need to be collected (including travel and immigration records; doctors, emergency, and hospital admissions; drug purchases; social network and location data; and even animal health data)
- Such data sets are **large**, **dynamic**, **complex**, **heterogeneous**, and **distributed**
- Privacy and confidentiality concerns arise if such data are stored and linked (at a central location)

Motivating example: LSD

- To tackle complex issues such as a global pandemic, we need to be able to **integrate** and **link large** and **complex**, and highly **sensitive** and **confidential** databases in near **real time**
- **Linking Sensitive Data** (LSD) is concerned with the development of methods, techniques, algorithms, and processes to achieve this goal
- Besides being a crucial tool in understanding a pandemic, LSD has applications in a variety of different domains (ranging from the health and social sciences to national censuses, crime and fraud detection, and national security)

Outline

- History and application examples for LSD
  - A brief history of linking databases
  - Modern applications of where LSD is required
  - Challenges for LSD
- Some key technologies used for LSD
  - How to compare names and addresses?
  - How to encrypt and encode sensitive data?
  - How to compare encrypted and encoded data?
  - Privacy-preserving record linkage
- Vulnerabilities of sensitive data (Anushka)
- Conclusions and research directions

Linking data is nothing new...

‘Linking’ London underground tickets to conduct traffic analysis in 1936

The book of life (Halbert Dunn, 1946)

- The idea of creating a book of life for each individual by linking records from birth, marriage, and death certificates, as well as records about individuals from the health and social security systems
- Each such book would start with a birth and end with a death record

- Dunn saw that linked records can provide a wealth of of information that is not available otherwise
- He also realised the challenges of **data quality**, large **volumes of data**, and **sensitivity** of personal data

Computer-based record linkage

- Computer assisted record linkage goes back as far as the 1950s (based on ad-hoc heuristic methods)
- Theoretical foundation for **probabilistic record linkage** by statisticians Fellegi and Sunter (1969)
  - No unique entity identifiers available (no person numbers or patient identifiers)
  - Compare names, addresses, dates of birth, and so on
  - Assign different importance to different such fields (same name is more important than same gender)
  - Classify a compared record pair as a **match**, a **non-match**, or a **potential match**
- Still the basis of many record linkage systems

Enter computer science...

- Strong interest in the last two decades from computer science (from research fields including data mining, AI, knowledge engineering, information retrieval, information systems, databases, and digital libraries)
- Many different techniques have been developed
- Major focus has been on scalability to very large databases and improving linkage quality
  - Blocking techniques to efficiently and effectively generate candidate record pairs
  - Machine learning-based classification techniques
- Development of **privacy-preserving record linkage** (PPRL) techniques

Applications of record linkage

- Remove duplicates in one data set (deduplication)
- Merge new records into a larger master data set
- Create patient or customer oriented statistics
  (for example for longitudinal studies)
- Clean and enrich data for analysis and mining
- Geocode matching (to facilitate spatial data analysis)
- Widespread use of record linkage
  - Health and social science research
  - Immigration, taxation, social security, national censuses
  - Business mailing lists, consumer product matching
  - Crime and fraud detection, and terrorism intelligence

Application examples

- Linking health records between hospitals
  - For health studies to obtain the full histories of patients
  - To identify patients who have visited multiple hospitals
  - Hospitals are unlikely to openly share their databases
  - Requires the linking of highly sensitive data potentially across hundreds of hospitals
- Linking census data over time
  - Many countries conduct censuses on a regular basis
  - To create longitudinal data about a population, census data need to be linked over time
  - Challenging due to changes in personal details, such as names and addresses
  - Laws might prohibit storing personal data over time

Major challenges when linking data

- No unique entity identifiers are available
- Real world data are dirty

  (typographical errors and variations, missing and outdated values, and various other data quality issues)
- Scalability to linking large databases
  - Naive comparison of all record pairs does not scale
- No ground truth data (gold standard) in many linkage applications
  - No record pairs with known true match status
- Privacy and confidentiality

  (because personal information, such as names and addresses, are commonly required for linking)

Outline

- History and application examples for LSD
  - A brief history of linking databases
  - Modern applications of where LSD is required
  - Challenges for LSD
- Some key technologies used for LSD
  - How to compare names and addresses?
  - How to encrypt and encode sensitive data?
  - How to compare encrypted and encoded data?
  - Privacy-preserving record linkage
- Vulnerabilities of sensitive data (Anushka)
- Conclusions and research directions

Comparing names and addresses

- A key requirement to achieve high linkage quality
- Personal data is prone to errors and variations
  - Scanned, hand-written, over telephone, hand-typed
  - Different correct spelling variations for proper names (*Christopher*, *Kristopher*, *Christoffer*, *Christophir*, *Christoph*, *Kristoffe*, *Christophe*, and many more..)
  - Nicknames (*Tash* for *Natasha* or *Tosh* for *Macintosh*)
  - Fake values (my phone number is *+61 04 1234 5678*)
- Changes occur over time (names can change due to marriage, and addresses when people move)
- Therefore, exact comparisons of names and addresses will not give good linkage results

Approximate name comparison

- Aim: Compare two names (or addresses) and calculate a numerical similarity between 0 and 1
  - Comparing a name with itself gives a similarity of 1 (compare *Peter* with *Peter*)
  - Comparing completely different names gives a similarity of 0 (compare *Peter* with *David*)
  - Comparing somewhat similar names gives a similarity between 0 and 1 (compare *Peter* with *Petros*)
- Many different techniques have been developed, some specific to names, others for general text

  (comparing text is a fundamental aspect in many applications, such as Web search, NLP, bioinformatics, spell checking, and many more)

Q-gram based name comparison

- Convert a name into *q-grams* (segments of length *q*)
  - For example, for *q* = 2: *peter* $\rightarrow$ [**pe**, **et**, te, er]
    petros $\rightarrow$ [**pe**, **et**, tr, ro, os]
- Find how many q-grams are common between two names (for our example, two: [pe, et])
- Calculate a similarity, for example using the Sørensen-Dice coefficient (developed by botanists in the 1940s to calculate the similarity between plant communities)

$$sim = 2 \cdot 2 / (4 + 5) = 4 / 9 = 0.44$$

- The more q-grams two names have in common the higher their similarity is

Encoding sensitive data

- We cannot share sensitive data (such as personal information) between organisations
- We need to encode and/or encrypt sensitive data
- PPRL employs techniques such as those used for secure Internet communication (like online banking)
- One key technique is *secure one-way hashing*
  - A function that converts an input into a hash code
  - If we only have the code then it is almost impossible to obtain the input
  - For example: *peter* $\rightarrow$ *4R#x+Y4i9!e@t4o]W*
    petros $\rightarrow$ *Z5%o-(7Tq1g?7iE/#*
- But this only allows for exact matching!

Privacy-preserving record linkage (1)

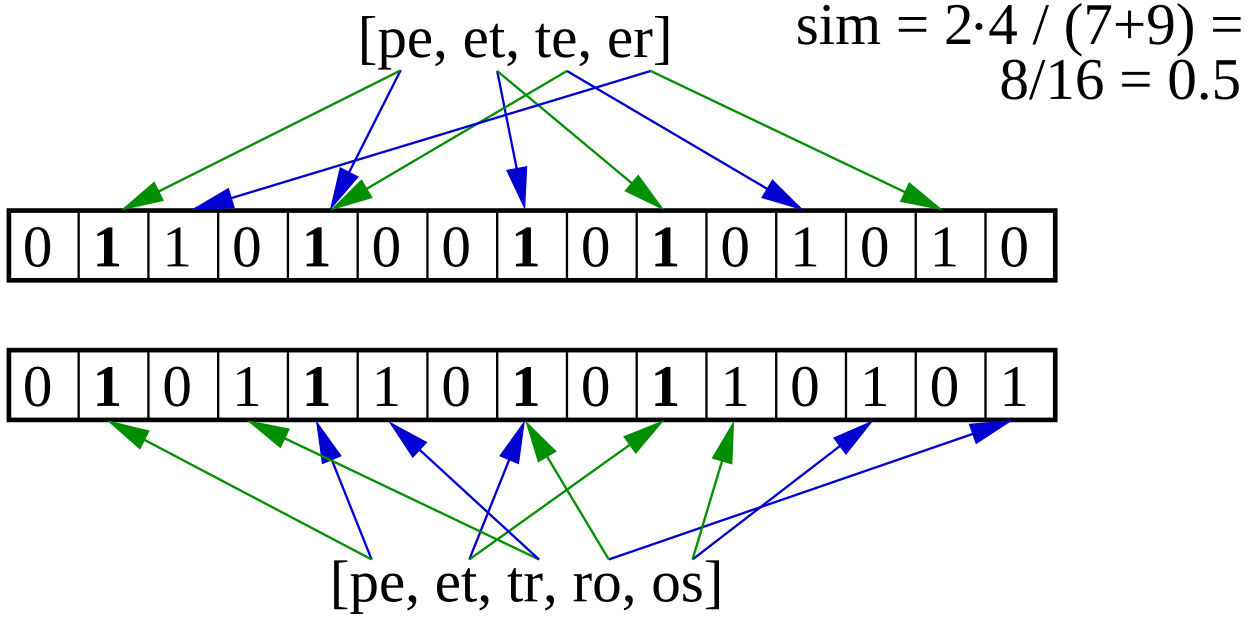
- We aim to link records in databases across organisations without revealing any sensitive data
- We require techniques that:
  - Allow for approximate matching and high linkage quality
  - Are provably secure (cannot be attacked) and do not allow the re-identification of encoded sensitive values
  - Are scalable to linking very large databases
- An active area of research since the mid 1990s
  - Contributions from computer science, statistics, as well as the health and social sciences
  - Besides the health domain, there is increasing interest by governments (such as for national censuses and digital vaccination passports)

Privacy-preserving record linkage (2)

- PPRL techniques can be categorised into secure multiparty computation (SMC) and perturbation based techniques
  - SMC based techniques are provably secure but have generally higher computation and communication requirements
  - Perturbation based techniques are more efficient, allow for approximate matching (of different types of data), and they are easier to implement
- However, perturbation based techniques are generally more vulnerable to privacy attacks than SMC techniques

Bloom filter based encoding

- Bloom filters were developed in 1970; their use for LSD was proposed by Rainer Schnell in 2009
- We *map* q-grams into bit arrays (0s and 1s) where the number of common 1-bits approximates the similarity



- Basic Bloom filters can be susceptible to attacks aimed at re-identifying sensitive values

Evaluating PPRL techniques

- Traditional evaluation of linkage techniques only considers linkage quality and scalability
  - Quality measures such as precision, recall, sensitivity, the F*-measure, positive predictive value, and others (if ground truth data is available)
  - Scalability measures such as reduction ratio, run-time, memory usage, etc.
- Evaluating privacy is more challenging
  - No single measure for privacy
  - Measures from statistical disclosure control or information theory have been adapted
  - Recent work is looking at **vulnerability assessments**

A Vulnerability Assessment Framework for Privacy-Preserving Record Linkage (PPRL)

Anushka Vidanage[1], Peter Christen[1], Thilina Ranbaduge[2], and Rainer Schnell[3]

[1] School of Computing
College of Engineering and Computer Science
The Australian National University
Canberra, Australia

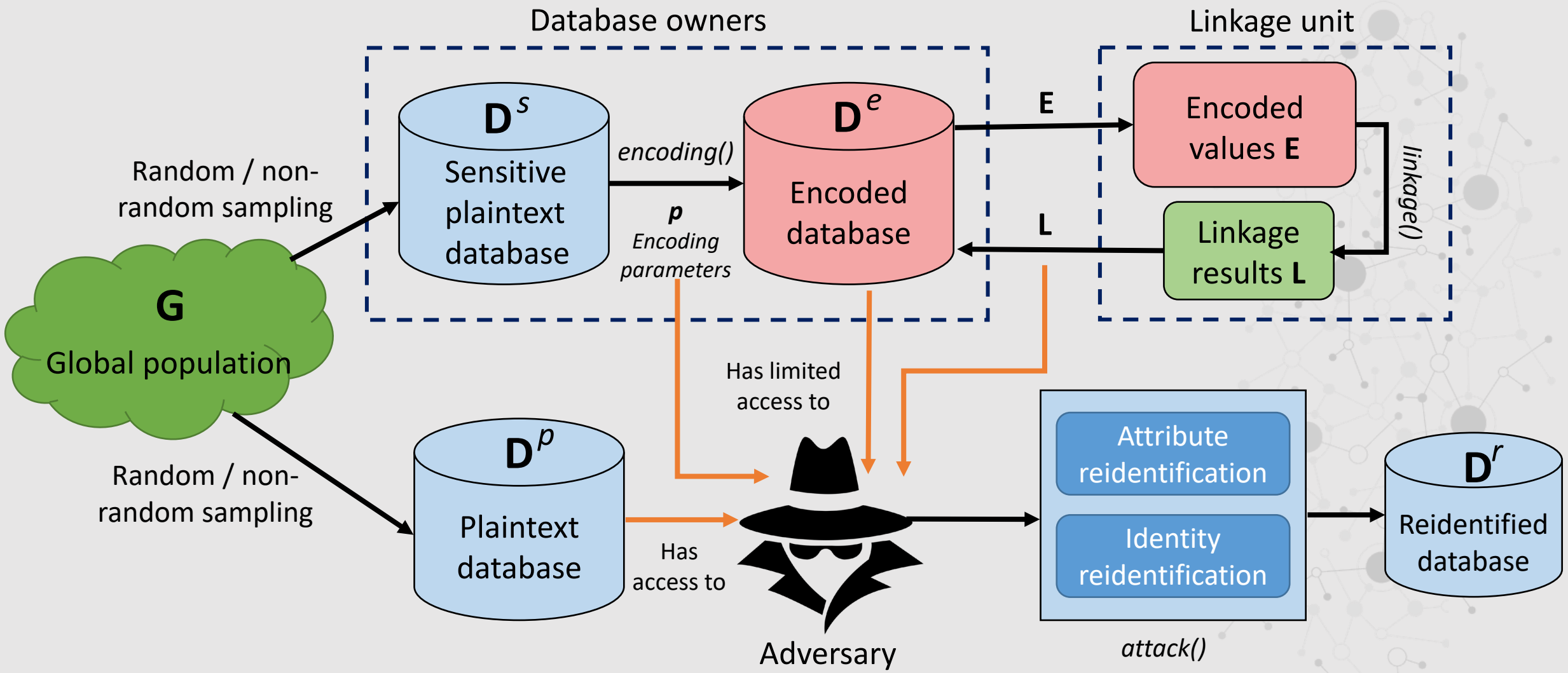
[2] Data61, CSIRO,
Canberra, Australia

[3] Methodology Research Group
University Duisburg-Essen
Duisburg, Germany

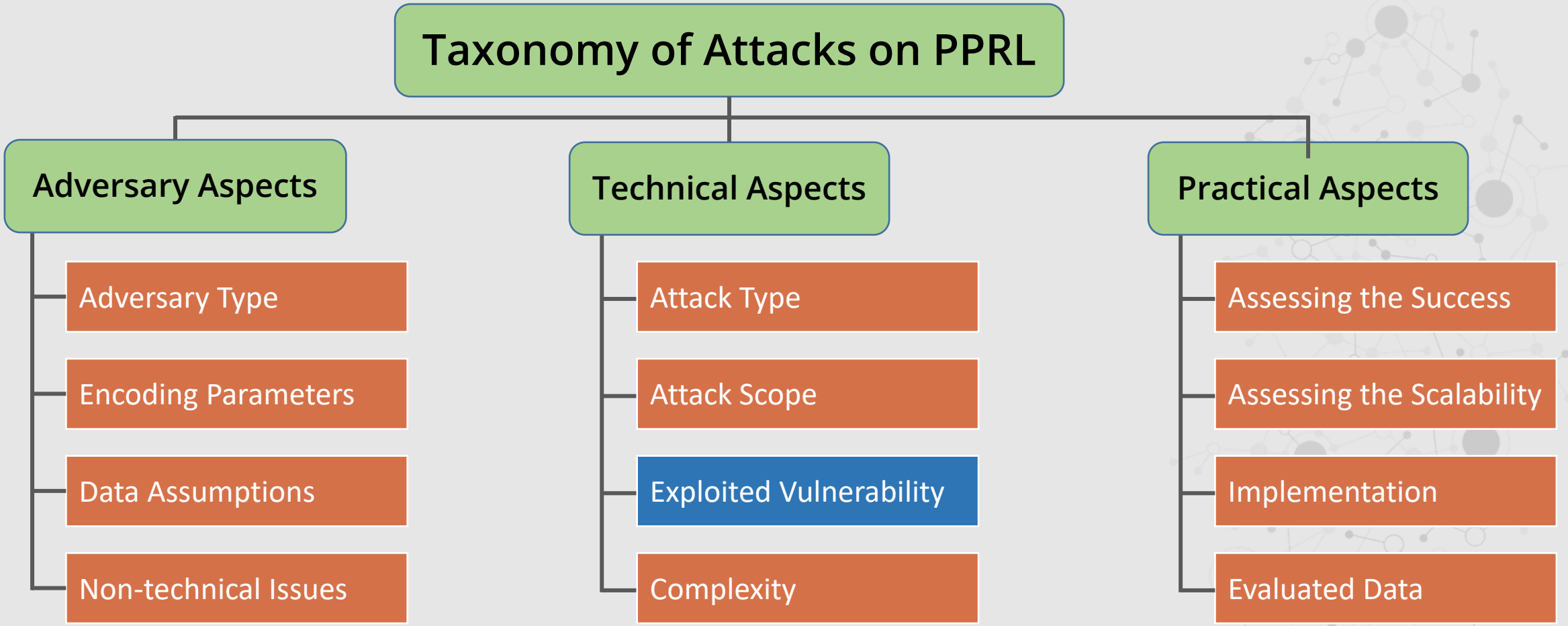
Contact: anushka.vidanage@anu.edu.au, peter.christen@anu.edu.au

Code: https://github.com/anushkavidanage/pprlVulnerabilityAnalysis

Overview of an Attack on PPRL



A Taxonomy of Attacks on PPRL



A Taxonomy of Attacks on Privacy-preserving Record Linkage: Anushka Vidanage, Thilina Ranbaduge, Peter Christen, and Rainer Schnell, *Journal of Privacy and Confidentiality,* 2022 (accepted).

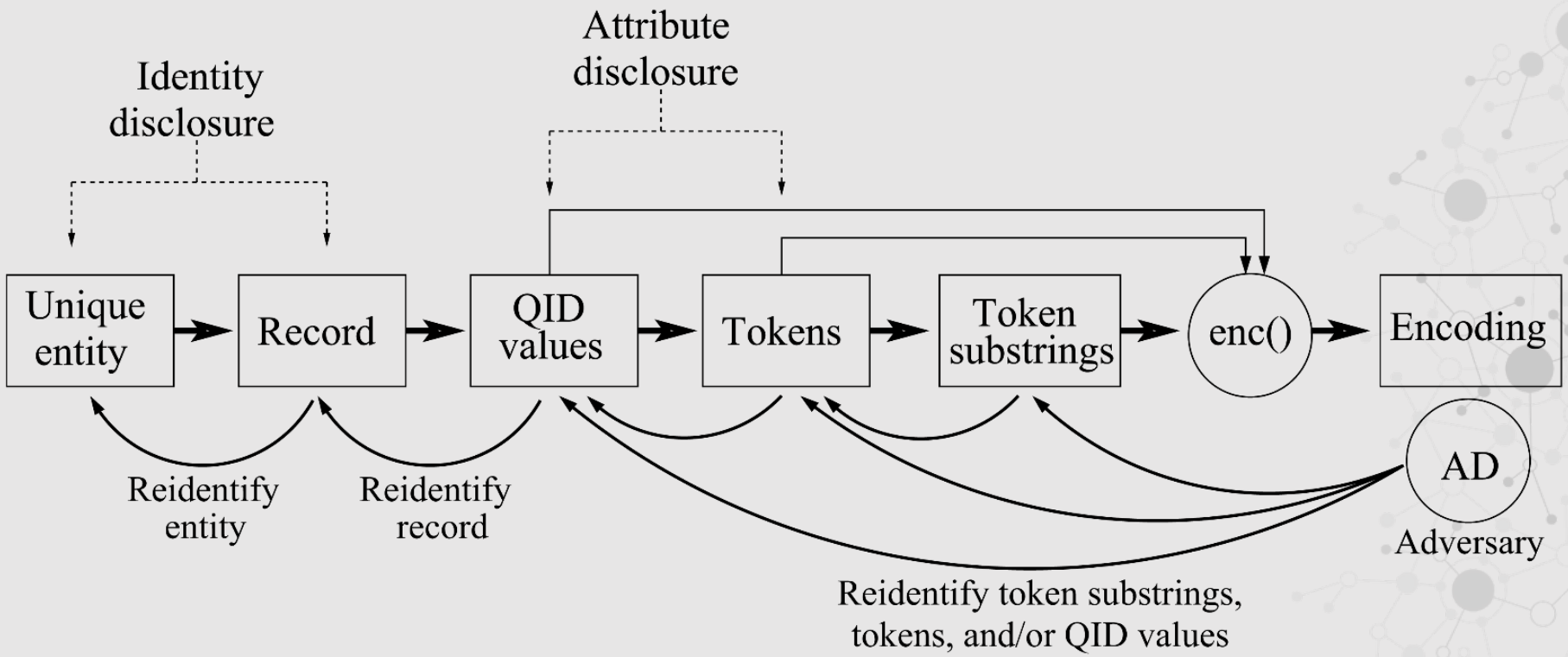
A Vulnerability Assessment Framework for PPRL

	First name	Last name	Street address	City
Record	Jean Pierre	Miller	42 Miller Street	Chapel Hill
QID values (Q)	Jean Pierre, Miller, 42 Miller Street, Chapel Hill			
Tokens (T)	Jean, Pierre, Miller, 42, Street, Chapel, Hill			
Token substrings (S)	an, ap, ch, ea, ee, el, er, et, ha, hi, ie, il, je, le, ll, mi, pe, pi, re, rr, st, tr, 42			
BF encoding[1] (Schnell et al. 2009)	0 1 1 0 0 1 0 1 1 1 0 1 0 1 0 1 0 1 0 0 0 1 1 1 0 1 1 1 0 1 0 0 1 1 1 0 1 1 0 1			
TMH Encoding[2] (Smith 2017)	1 0 0 0 1 0 1 0 1 0 1 1 0 1 0 1 1 0 0 1 1 1 0 1 1 0 0 1 1 0 1 0 1 0 1 1 0 1 0 0			
MMK Encoding[3] (Randall et al. 2019)	i+sL9RtXd4Jb, avxxlTxIoLx3, Lr1dnWLGM/K8			
2SH Encoding[4] (Ranbaduge et al. 2020)	[128, 2, 8, 143, 16, 146, 148, 26, 155, 28, 34, 37, 166, 168, 45]			
SLK Encoding[5] (Karmel 2005)	SHA-2(ileea150218871) (assuming date of birth as 15.02.1887 and gender as male)			

- A record can be converted into set of QID values (q_i), a set of tokens (t_i), and a set of token substrings (s_i)

1. Bloom filter encoding
2. Tabulation min-hash encoding
3. Multiple dynamic match-key encoding
5. Two-step hash encoding
5. Statistical linkage key

A Vulnerability Assessment Framework for PPRL



- An adversary usually aims to reverse-engineer the encoding process to reidentify encoded values
- Hence, either token substrings, tokens, or QID values should be vulnerable

Vulnerability of a Single Value

- Three types of vulnerabilities of a single value can be exploited by an attack

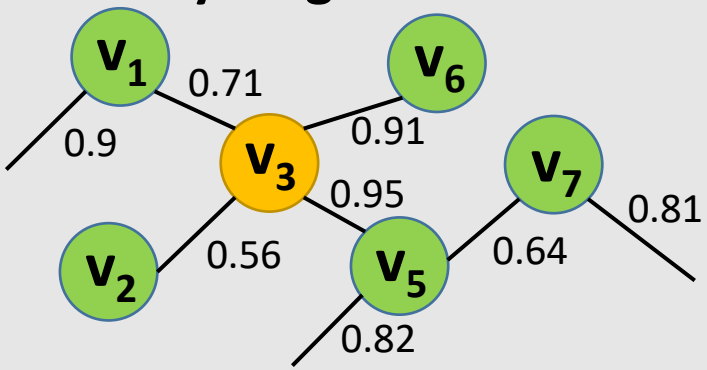
Frequency vulnerability

Last name	Frequency
Smith	800
Johnson	400
David	395
Vidanage	2

Length vulnerability

Last name	Length
Wolfeschlegelsteinhausenbergerdorff[1]	35
Kellermann	10
Williams	8
Li	2

Similarity neighbourhood vulnerability



1 https://en.wikipedia.org/wiki/Hubert_Blaine_Wolfeschlegelsteinhausenbergerdorff_Sr.

Vulnerability of a Pair of Values

- Two types of vulnerabilities of a value pair can be exploited by an attack on PPRL

Co-occurrence vulnerability

Value/ Value pair	Frequency
Johnson	400
David	395
David, Johnson	150
Peter, Miller	90

Similarity vulnerability

Value pair	Edit distance similarity
Miller, Mills	0.67
Smith, Smyth	0.8
Johnson, John	0.57
Peter, Pete	0.8

Vulnerability Conditions

- A plaintext and an encoded value need to satisfy one and two conditions, respectively, in order for them to be vulnerable in their corresponding databases

Definition 1. $(\varepsilon, k) - vulnerability$ (for plaintext and encoded values)
For a value $v_i \in \mathbf{V}$ and the set $\mathbf{v}_i = \{v_j : func(v_i, v_j) \leq \varepsilon, v_i \neq v_j\}$ of other values $v_j \in \mathbf{V}$ with a tolerance $\varepsilon \geq 0$, we define v_i as $(\varepsilon, k) - vulnerable$ in the database with regard to the function $func()$ if $0 \leq |\mathbf{v}_i| < k$.

Definition 2. $(\varepsilon, k) - assignability$ (for encoded values)
For an encoding $e_i \in \mathbf{E}$ that is $(\varepsilon, k) - vulnerable$ within the set $\mathbf{E}$, we define a set $\mathbf{m}_i = \{v_a : v_a \in \mathbf{V}, func(e_i, v_a) \leq \varepsilon\}$ of vulnerable plaintext values $v_a \in \mathbf{V}$ that can be assigned to the distinguishable encoding e_i based on a function $func()$ and a tolerance ε. If $1 \leq |\mathbf{m}_i| < k$, with k and $\varepsilon \geq 0$ being privacy parameters, then we define the pair (e_i, v_a) as $(\varepsilon, k) - assignable$.

Vulnerability Analysis (1)

- Analysis of existing attack methods on PPRL

✓ Exploited vulnerability

Attack method	Frequency	Length	Co-occurrence	Similarity	Similarity neighbourhood
Constraint satisfaction based attack *(Kuzu et al. 2011)*	✓	✓	✓		
Frequency based manual attack *(Niedermeyer et al. 2014)*	✓				
Optimisation based attack *(Kroll and Steinmetzer 2015)*	✓		✓		
Frequency based attack *(Christen et al. 2017)*	✓				
Graph traversal based attack *(Mitchell et al. 2017)*			✓		
Graph matching based attack *(Culnane et al. 2017)*					✓
Frequency based attack *(Christen et al. 2018a)*	✓	✓			
Pattern-mining based attack *(Christen et al. 2018b)*	✓		✓		
Frequency analysis attack *(Vidanage et al. 2020a)*	✓				
Graph matching based attack *(Vidanage et al. 2020b)*					✓

Vulnerability Analysis (2)

Analysis of PPRL encoding techniques

Vulnerability	Plaintext			Encoding				
	QID	Token	Substring	SLK[1]	BF[2]	TMH[3]	MMK[4]	2SH[5]
Frequency	✓	✓	✓	✓	✓	✓	✓	✓
Length	✓	✓			✓	?		?
Co-occurrence	✓	✓	✓		✓	?	✓	?
Similarity	✓	✓			?	?	?	?
Similarity neighbourhood	✓	✓			✓	✓	?	✓

✓ Exploited vulnerability ? Potential vulnerability

1. Statistical linkage key (Karmel 2005)
2. Bloom filter encoding (Schnell et al. 2009)
3. Tabulation min-hash encoding (Smith 2017)
4. Multiple dynamic match-key encoding (Randall et al. 2019)
5. Two-step hash encoding (Ranbaduge et al. 2020)

Experimental Evaluation

Databases

- North Carolina voter registration (NCVR)
  - Snapshot collected in December 2020
  - Randomly sampled a subset of 100,000 records
  - First name, Last name, Street address, City
- European census database (EURO)
  - 25,343 fictitious records
  - First name, Last name, Street address, City

Evaluation criteria

- Analysed all five vulnerabilities for the three plaintext value types (QIDs, tokens, and token substrings)
- Analysed five encoding technique BF, TMH, MMK, 2SH, and SLK
- We used k = [10, 20] and ε = [0%, 1%] for the privacy parameter settings

Plaintext Vulnerability Results

Percentages of how many values are vulnerable

		First Name				City			
		k =	10		20		10		20	
		ε =	0%	1%	0%	1%	0%	1%	0%	1%
NCVR	Freq	**QID**	3.42	0.38	4.64	0.67	67.9	2.76	100	5.80
		Token	3.34	0.41	4.70	0.68	70.9	2.92	97.3	7.17
		Substring	65.1	10.6	74.4	12.9	100	13.0	100	22.4
	Len	**QID**	0.06	0.06	0.40	0.40	3.18	3.18	3.18	3.18
		Token	0.18	0.18	0.18	0.18	0.93	0.93	7.04	7.04
	Co-occur	**QID**	-	-	-	-	-	-	-	-
		Token	4.85	4.85	4.85	4.85	95.2	10.0	100	12.9
		Substring	5.97	0.19	8.68	0.54	18.8	0.40	28.9	0.77
	Sim	**QID**	0.02	0	0.03	0.01	7.35	2.0	16.7	3.65
		Token	0.01	0	0.03	0.01	3.65	1.77	8.43	4.26
	Sim Neigh	**QID**	100	63.0	100	84.9	100	100	100	100
		Token	100	64.3	100	85.9	100	100	100	100

Encoded Vulnerability Results

Percentages of how many values are vulnerable

		First Name				First name, Last name, Street address, City			
		k =	10		20		10		20	
		ε =	0%	1%	0%	1%	0%	1%	0%	1%
NCVR	Freq	**BF**	3.43	0.38	4.67	0.67	0.01	0.01	0.01	0.01
		TMH	3.43	0.38	4.67	0.67	0.01	0.01	0.01	0.01
		2SH	3.43	0.38	4.67	0.67	0.01	0.01	0.01	0.01
	Len	**BF**	1.78	0.30	3.63	0.62	0.25	0.03	0.58	0.06
		TMH	0.88	0.25	1.31	0.55	0.08	0.02	0.14	0.04
		2SH	1.62	0.30	2.99	0.53	0.30	0.02	0.56	0.04
	Co-occur	**BF**	-	-	-	-	0.02	0.01	0.04	0.01
	Sim	**BF**	1.65	0	3.26	0.01	0.08	0	0.15	0
		TMH	0.01	0	0.02	0	0.01	0	0.01	0
		2SH	1.4	0	2.62	0	0.10	0	0.17	0
	Sim Neigh	**BF**	100	49.3	100	70.6	99.9	4.3	99.9	8.4
		TMH	69.3	17.5	87.5	25.5	99.9	47.1	99.9	59.3
		2SH	100	47.2	100	77.4	100	74.5	100	80.2

Encoded Vulnerability Results

Percentages of how many values are vulnerable

				SLK	MMK		
					First name, Street	Street, City	First name, Last name, Street
NCVR	Freq	k =	ε =				
		10	0%	0	0	0.04	0.01
			1%	0	0	0.02	0.01
		20	0%	0.2	0.02	0.04	0.01
			1%	0.2	0.02	0.03	0.01

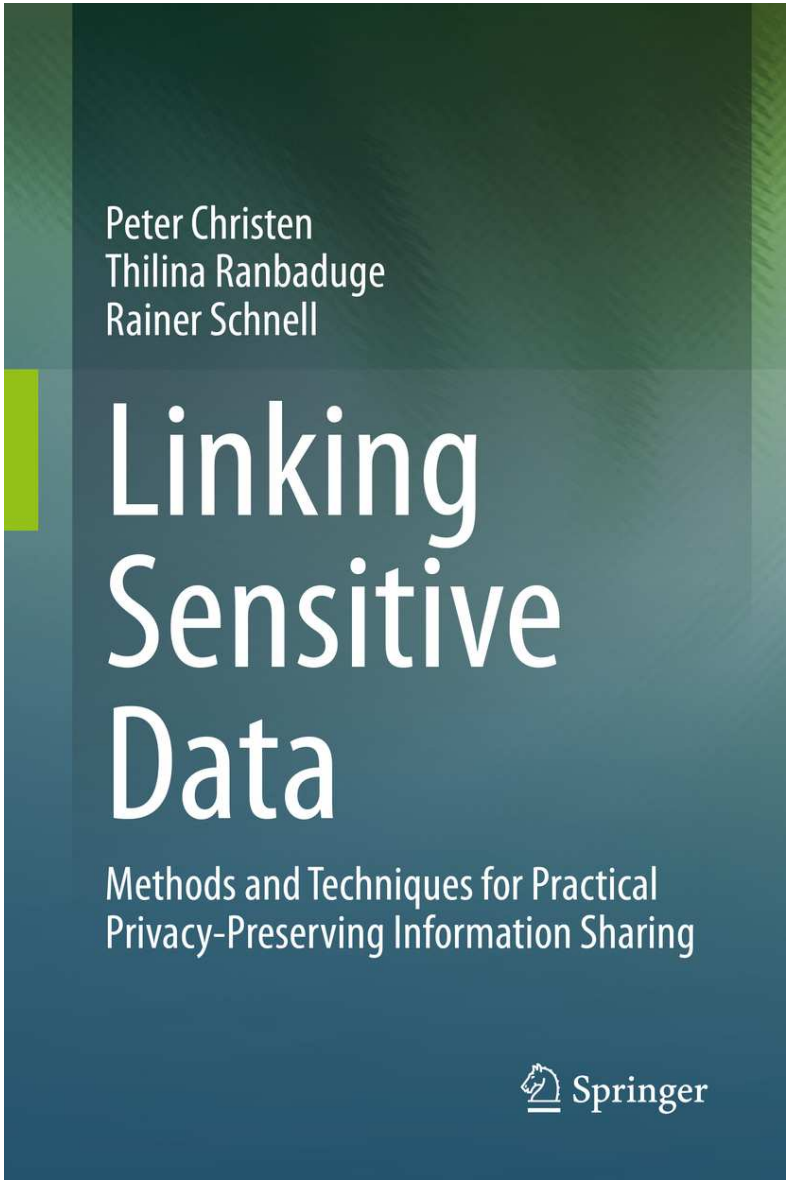
Practical aspects of PPRL

- From a practical perspective, various aspects of PPRL will be of importance
  - Formal legal constraints and their implementation
  - Dealing with dirty, missing, temporal, and dynamic data
  - Dealing with bias and uncertainty in linked data
  - Costs of false and missed matches
  - Lack of ground truth data, and how to evaluate linkage quality in a PPRL context
  - Suitability of a PPRL techniques for a given linkage scenario (how many communication steps needed)
  - The actual linkage scenario (including threat scenario)
  - Technical knowledge available in an organisation
  - Availability of software or ease of implementation

Conclusions and research directions

- The technical building blocks for LSD exist, and we can now link large sensitive databases in privacy-preserving ways
- There are various open questions and challenges
  - How do we securely link new types of data, such as images, biometrics, or genetic data?
  - How do we evaluate a linkage if only encoded or encrypted values are available?
  - How do we prove our techniques are secure and cannot be attacked? Who are the adversaries?
  - How do we measure and formalise privacy?
  - How to do real-time linking of very large and dynamic databases?

'Linking Sensitive Data' book
(Springer, Nov 2020)



The Book describes how linkage methods work and how to evaluate their performance.
It covers all the major concepts and methods and also discusses practical matters such as computational efficiency, which are critical if the methods are to be used in practice – and it does all this in a highly accessible way!

Prof David J. Hand OBE,
Imperial College, London