Privacy-Preserving Data Sharing and Matching

Peter Christen

School of Computer Science,
ANU College of Engineering and Computer Science,
The Australian National University,
Canberra, Australia

Contact: **peter.christen@anu.edu.au**

Project Web site: **http://datamining.anu.edu.au/linkage.html**

THE AUSTRALIAN NATIONAL UNIVERSITY

Outline

- Short introduction to data sharing and matching
  - Applications, techniques and challenges
- Privacy and confidentiality issues with data sharing and matching
- Data sharing and matching scenarios
  - Illustrate privacy and confidentiality issues
- Privacy-preserving sharing and matching approaches
  - *Blindfolded data linkage* in more details
- Challenges and research directions

Data sharing

- Data(bases) that contain personal or confidential information are often distributed
  - Vertically-partitioned: Different attributes in different organisations

    For example:*Centrelink* $\longleftrightarrow$ *Medicare*
  - Horizontally-partitioned: Different records in different organisations

    For example: *NSW Health* $\longleftrightarrow$ *QLD Health*
- Question: How to conduct data analysis on combined data(bases) without having to exchange (and thus reveal) private or confidential data between organisations?

Data matching

- The process of matching and aggregating records that represent the same entity (such as a patient, a customer, a business, an address, an article, etc.)
  - Also called *data linkage*, *entity resolution*, *data scrubbing*, *object identification*, *merge-purge*, etc.
- Challenging if no unique entity identifiers available

  For example, which of these three records refer to the same person?

Dr Smith, Peter	*42 Miller Street 2602 O'Connor*
Pete Smith	*42 Miller St, 2600 Canberra A.C.T.*
P. Smithers	*24 Mill Street; Canberra ACT 2600*

Applications of data matching

- Health, biomedical and social sciences
  (for epidemiological or longitudinal studies)
- Census, taxation, immigration, and social security
  (for improved data processing and analysis)
- Deduplication of (business mailing) lists
  (to improve data quality and reduce costs)
- Crime and fraud detection, national security
- Geocode matching (‘geocoding’) of addresses to locations for spatial analysis
- Bibliographic databases and online libraries
  (to measure impact - for example for *ERA*)

Data matching techniques

- Deterministic matching
  - Exact matching (if a *unique identifier* of high quality is available: precise, robust, stable over time)
    Examples: *Medicare*, *ABN* or *TFN* (?)
  - Rules based matching (complex to build and maintain)
- Probabilistic matching
  - Use available (personal) information for matching (like *names*, *addresses*, *dates-of-birth*, etc.)
  - Can be wrong, missing, coded differently, or out-of-date
- Modern approaches
  (based on machine learning, AI, data mining, database, or information retrieval techniques)

Data matching challenges

- Real world data is dirty
  (typographical errors and variations, missing and out-of-date values, different coding schemes, etc.)
- Scalability
  - Comparison of all record pairs has quadratic complexity (however, the maximum number of matches is in the order of the number of records in the databases)
  - Some form of blocking, indexing or filtering required
- No training data in many matching applications
  - No record pairs with known true match status
  - Possible to manually prepare training data (but, how accurate will manual classification be?)

Privacy and confidentiality issues

- The public is worried about their information being shared and matched between organisations
  - Good: health and social research; statistics, crime and fraud detection (taxation, social security, etc.)
  - Scary: intelligence, surveillance, commercial data mining (not much details known, no regulation)
  - Bad: identity fraud, re-identification
- Traditionally, *identified data* has to be given to the person or organisation performing the matching
  - Privacy of individuals in data sets is invaded
  - Consent of individuals needed (often not possible, so approval from ethics review boards required)

Data sharing scenario

- Two pharmaceutical companies are interested in collaborating on the development of new drugs
- The companies wish to identify how much overlap of confidential data there is in their databases
  (without having to reveal any of that data to each other)
- Techniques are required that allow comparison of large amounts of data such that similar data items are found (while all other data is kept confidential)
- Involvement of a third party to undertake the matching will be undesirable
  (due to the risk of collusion of the third party with either company, or potential security breaches at the third party)

Data matching scenario (1)

- A researcher is interested in analysing the effects of car accidents upon the health system
  - *Most common types of injuries?*
  - *Financial burden upon the public health system?*
  - *General health of people after they were involved in a serious car accident?*
- She needs access to data from hospitals, doctors, car insurances, and from the police
  - All identifying data has to be given to the researcher, or alternatively a trusted data matching unit
- This might prevent an organisation from being able or willing to participate (car insurances or police)

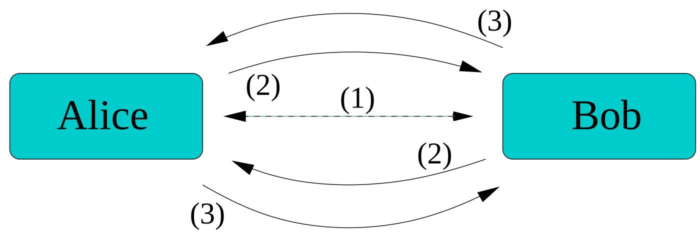
Data matching scenario (2)

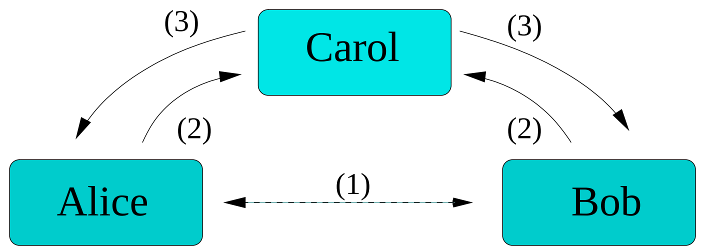
- A researcher has access to several de-identified data sets (which separately do not permit individuals to be re-identified)
- He has access to a HIV database and a midwives data set (both contain postcodes, and year and month of birth – in the midwives data for both mothers and babies)
- Using birth notifications from a public Web site (news paper), the curious researcher is able to match records and identify births in rural areas by mothers who are in the HIV database
- Re-identification is a big issue due to the increase of data publicly available on the Internet

Geocode matching scenario

- A cancer register aims to geocode its data
  (to conduct spatial analysis of different types of cancer)
- Due to limited resources the register cannot invest in an in-house geocoding system
  (software and personnel)
- They are reliant on an external geocoding service
  (commercial geocoding company or data matching unit)
- Regulations might not allow the cancer register to send their data to any external organisation
- Even if allowed, complete trust is required into the geocoding service (to conduct accurate matching, and to properly destroy the register's address data afterwards)

Privacy-preserving sharing and matching approaches

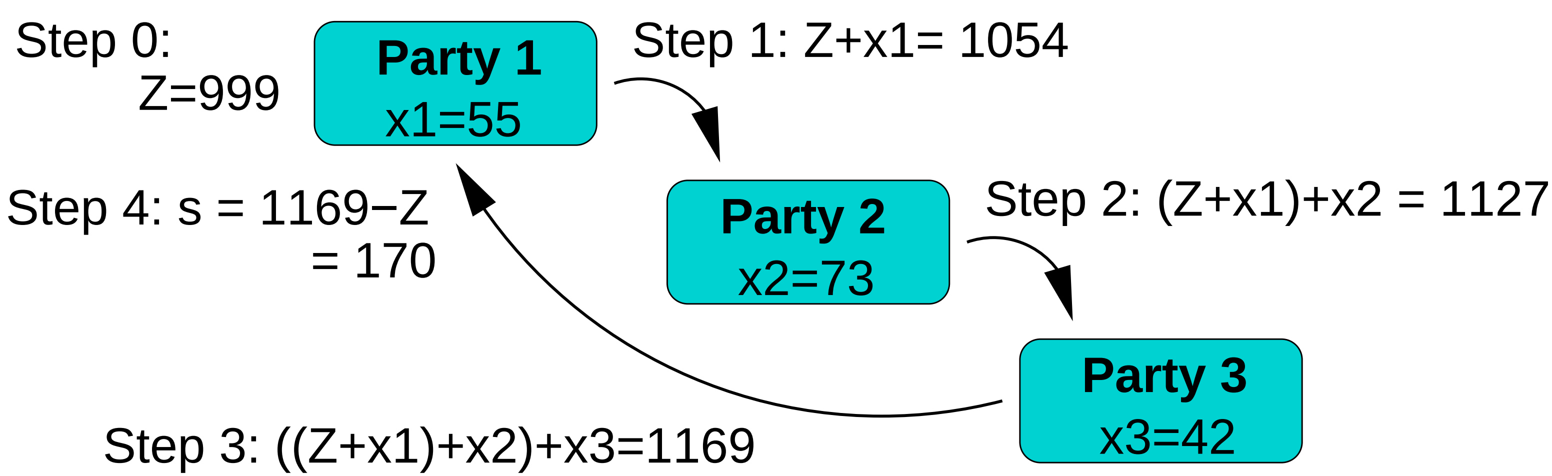




- Based on cryptographic techniques
  (secure multi-party computations – *more on next slide*)
- Assume two data sources, and possibly a third (trusted) party to conduct the matching
- Objective: No party learns about the other parties' private data, only matched records are released
  - Various approaches with different assumptions about threats, what can be inferred by parties, and what is being released

Secure multi-party computation

- Compute a function across several parties, such that no party learns the information from the other parties, but all receive the final results *[Yao 1982; Goldreich 1998/2002]*
- Simple example: Secure summation $s = \sum_i x_i$.



Privacy-preserving matching techniques

- Pioneered by French researchers for exact matching *[Dusserre et al. 1995; Quantin et al. 1998]*
  - Using one-way hash-encoding (*'tim'* $\rightarrow$ *'51d3a6a70'*)
- Secure and private sequence comparisons (edit distance) *[Atallah et al. WPES'03]*
- Blindfolded record linkage (details on following slides) *[Churches and Christen, BioMed Central 2004]*
- Secure protocol for computing string distance metrics (TF-IDF and Euclidean distance) *[Ravikumar et al. PSDM'04]*
- Privacy-preserving blocking *[Al-Lawati et al. IQIS'05]*

Blindfolded data linkage

- Based on approximate string matching using hash-encoded *q*-grams
- Assuming a three-party protocol
  - Alice has database **A**, with attributes **A.a**, **A.b**, etc.
  - Bob has database **B**, with attributes **B.a**, **B.b**, etc.
- Alice and Bob wish to determine whether any of the values in **A.a** match any of the values in **B.a**, without revealing the actual values in **A.a** and **B.a**
- Easy if only *exact matches* are considered
- More complicated if values contain errors or variations (a single character difference between two strings will result in very different hash codes)

Protocol – Step 1

- A protocol is required which permits the *blind* calculation by a trusted third party (Carol) of a more general and robust measure of similarity between pairs of secret strings
- Proposed protocol is based on *q*-grams
  For example (q = 2, bigrams): *'peter'* $\rightarrow$ *('pe','et','te','er')*
- Protocol step 1
  - Alice and Bob agree on a secret random key
  - They also agree on a secure one-way message authentication algorithm (HMAC)
  - They also agree on a standard of preprocessing strings

Protocol – Step 2

- Protocol step 2
  - Alice computes a sorted list of *q*-grams for each of her values in **A.a**
  - Next she calculates all non-empty sorted *q*-gram sub-lists (power-set without empty set)
    For example: *‘peter’* $\rightarrow$ *[(‘er’), (‘et’), (‘pe’), (‘te’), (‘er’,‘et’), (‘er’,‘pe’), (‘er’,‘te’), (‘et’,‘pe’), (‘et’,‘te’), (‘pe’,‘te’), (‘er’,‘et’,‘pe’), (‘er’,‘et’,‘te’), (‘er’,‘pe’,‘te’), (‘et’,‘pe’,‘te’), (‘er’,‘et’,‘pe’,‘te’)]*
  - Then she transforms each sub-list into a secure hash digest and stores these in **A.a_hash_bigr_comb**

Protocol – Steps 2 and 3

- Protocol step 2 (continued)
  - Alice computes an encrypted version of the record identifier and stores it in **A.a_encrypt_rec_key**
  - Next she places the number of bigrams of each **A.a_hash_bigr_comb** into **A.a_hash_bigr_comb_len**
  - She then places the length (total number of bigrams) of each original string into **A.a_len**
  - Alice then sends the quadruplet [**A.a_encrypt_rec_key**, **A.a_hash_bigr_comb**, **A.a_hash_bigr_comb_len**, **A.a_len**] to Carol
- Protocol step 3
  - Bob carries out the same as in step 2 with his **B.a**

Protocol – Step 4

- Protocol step 4
  - For each value of **a_hash_bigr_comb** shared by **A** and **B**, for each unique pairing of [**A.a_encrypt_rec_key**, **B.a_encrypt_rec_key**], Carol calculates a **bigr_score** similarity (Dice coefficient):

$$\mathbf{bigr\_score} = \frac{\mathbf{2} \times \mathbf{A.a\_hash\_bigr\_comb\_len}}{(\mathbf{A.a\_len} + \mathbf{B.a\_len})}$$

  - Carol then selects the maximum **bigr_score** for each pairing [**A.a_encrypt_rec_key**, **B.a_encrypt_rec_key**] and sends these results to Alice and Bob (or she only send the number of matches with a **bigr_score** above a certain similarity threshold)

Example

- Alice: *'peter'* $\rightarrow$ *[('er'), ... ('et','pe','te'), ...]*
  For bigram sub-list *('et','pe','te')*:
  – **A.a_hash_bigr_comb** = *'W5gO1@'*
  – **A.a_hash_bigr_comb_len** = *3*
  – **A.a_len** = *4*
  Alice sends to Carol: *['A-7D4W', 'W5gO1@', 3, 4]*
- Bob: *'pete'* $\rightarrow$ *[('er'), ... ('et','pe','te')]*
  For bigram sub-list *('et','pe','te')*:
  – **B.a_hash_bigr_comb** = *'W5gO1@'*
  – **B.a_hash_bigr_comb_len** = *3*
  – **B.a_len** = *3*
  Bob sends to Carol: *['B-T5YS', 'W5gO1@', 3, 3]*
- Carol calculates: **bigr_score** = $\mathbf{\frac{2 \times 3}{(4+3)} = \frac{6}{7} = 0.857}$

THE AUSTRALIAN NATIONAL UNIVERSITY

Full blindfolded data linkage

- Several attributes **a**, **b**, **c**, etc. can be compared independently (by different Carols)
- Different Carols send their results to another party (David), who forms a (sparse) matrix by joining the results
- The final *matching weight* for a record pair is calculated by summing individual **bigr_score**s
- David arrives at a set of *blindly linked records* (pairs of [**A.a_encrypt_rec_key**, **B.a_encrypt_rec_key**])
- Neither Carol nor David learn what records and values have been matched

Challenges with privacy-preserving matching

- Many secure multi-party computations are computationally very expensive
  - Some have large communication overheads
  - Not scalable to very large databases
- Not integrated with modern classification techniques (because only encoded values are available, unsupervised learning is required)
- Assessment of matching quality is problematic (not easy to verify if matched records correspond to true matches, and how many true matches were missed)
- Re-identification can still be a problem (if released records allow matching with external data)

Research directions (1)

- Secure matching
  - New and improved secure matching techniques (such as better approximate string comparison functions)
  - Reduce computational complexity and communication overheads of current approaches
  - Frameworks and test-beds for comparing and evaluating secure data matching techniques are needed
- Automated record pair classification
  - In secure three-party protocols, the matching party only sees encoded data (no manual clerical review possible)
  - How to modify unsupervised classification techniques so they can work on encoded data?

THE AUSTRALIAN NATIONAL UNIVERSITY

Research directions (2)

- Scalability / Computational issues
  - Techniques for distributed (between organisations) matching of very large data collections are needed
  - Combine secure matching and automated classification with distributed and high-performance computing
  - Also to be addressed: access protocols, fault tolerance, data distribution, charging policies, user interfaces, etc.
- Preventing re-identification
  - Make sure de-identified data that is matched with other (public) data does not allow re-identification
  - Various possible approaches, such as *micro-data confidentiality* and *k-anonymity*

Conclusions

- Scalable, accurate, automated and privacy-preserving data matching is currently not feasible
- Four main research directions
  1. Improved secure matching
  2. Automated record pair classification
  3. Scalability and computational issues
  4. Preventing re-identification
- Public acceptance of data sharing and matching is another major challenge
- For more information see project Web site

  (publications, talks, *Febrl* data linkage software)

  http://datamining.anu.edu.au/linkage.html