De-identifying an EHR Database – Anonymity, Correctness and Readability of the Medical Record

Kostas Pantazos, Soren Lauesen, Soren Lippert
Problem: Electronic Health Record data exist, but clinicians cannot conduct research using them due to confidentiality.

Possible solutions:
1. Clinicians can use test data.
2. Clinicians can use data approved by patients.
3. Clinicians can use de-identified data.
Several de-identification algorithms have been developed \cite{Berman2003, Gupta2004, Sweeney1996, Szarvas2007, Uzuner2007, Velupillai2007}.

Previous research focus mainly on:

- De-identifying structured data, not in free-text. \cite{Meystre2010}
- Preserving anonymity and medical correctness \cite{Meystre2010}
- De-identifying parts of a database, but none has de-identified a full EHR database.
De-identify a full Danish EHR database and ensure:

• **Anonymity**: Replace all the identifiers.

• **Medical correctness**: Preserve medical information.

• **Consistency**: Identical identifiers in original version are also identical in de-identified version. E.g.: Carla Jensen and John Jensen => Maria Sorensen and Peter Sorensen

• **Readability**: Replace the identifiers with meaningful real values.
A de-identification example

<table>
<thead>
<tr>
<th>Original ‘Patient’ table (structured data)</th>
<th>De-identified ‘Patient’ Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPR-number*</td>
<td>First name*</td>
</tr>
<tr>
<td>290210-1546</td>
<td>Carla</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Original ‘Medical Record Line’ table (structured data and free-text)</th>
<th>De-identified ‘Medical Record Line’ table (structured data and free-text)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPR-number*</td>
<td>Medical note**</td>
</tr>
<tr>
<td>290210-1546</td>
<td><em>Copenhagen Hospital, Carla Sorensen, CPR: 290210-1546, Visit date 21-09-2010...</em>”</td>
</tr>
<tr>
<td>290210-1546</td>
<td><em>Copenhagen Hospital, Visit date 22-09-2010...Carla visited Copenhagen hospital...</em>”</td>
</tr>
</tbody>
</table>

*Structured data, ** Unstructured data (free-text)
1. **Number ambiguity:**
   1. A phone number can also be interpreted as a Civil Registration number (CPR).

2. **Language ambiguity:**
   1. “Hans” is a Danish pronoun, but can also be a male first name.
   2. Medical eponymous names, city names and clinic names can also be person names: e.g. Aaron.

3. **Corrupted data**
   1. Invalid CPR numbers in structured data.
Manual process

1.a DATABASE INVESTIGATION
- Locate tables
- Locate table’s fields (identifiers)
- Categorize identifiers based on:
  - Type
    - Identifying variables
    - Quasi Identifiers
  - Replaced in
    - 1. Structured
    - 2. Free-text

1.b DATABASE IDENTIFIERS
- Create de-identification rules for each type of identifier.
- Identify potential ambiguous cases for each identifier.
- Create de-identification rules for ambiguous cases.

2. EXTERNAL IDENTIFIERS
- Identify Danish ambiguous names.
- Identify medical eponyms.
- Identify other useful names (i.e., hospitals, clinics, cities).

Automated process

3. ALGORITHM

Create lists of:
- 1. Database identifiers
- 2. External identifiers

Define a replacement for each identifier.

Replace identifiers in structured data and free-text using:
- Lists of named entities
- Rules for ambiguous cases
- Simple language analysis
Danish EHR database (12GB) with 437,164 patient records. Each record contains diagnoses, notes, lab data, etc.

1. Investigated 65 tables:
   - 22 tables with identifiers which may expose identity:
     - 9 tables with only structured data and 13 tables with free text.
   - 43 tables without any identifier.

2. Investigated table fields:
   - 9 identifiers (e.g. CPR-number, name, etc.)
   - 13 Quasi-identifiers (e.g. zip-code, city, hospital, etc.)
Lists of:

1. Ambiguous Danish names from:
   - Danmarks Statistik: 28,628 names
   - Database: 437,164 patients

   Using Danish dictionary of Microsoft Word:
   - 3,557 potential ambiguous names
   - 2,793 potential ambiguous names

   Using specific health care domain knowledge:
   - 83 ambiguous names
   - 1,952 ambiguous names
   - 682 ambiguous names (mainly abbreviations)

   Total: 2,717

2. Medical eponymous names:
   - 3,246 names were extracted from (Who named it, 2010).
Lists of names to replace identifiers:

1. Hospitals: 93 names  

2. Clinics: 219 names  
   [Sygehusvalg, 2010]

3. Streets: 25,429 names  
   [Post Danmark, 2010]

4. Zip-codes: 1,396  
   [Post Danmark, 2010]

5. Cities: 681 names  
   [Post Danmark, 2010]
Algorithm

1. Read Patient table
2. Calculate name frequency and create lists
3. Delete patient records (~25% were ambiguous names and corrupted data)
4. Create replacement lists
5. Update Patient table
6. Read other tables
7. Alter structured data and free text
Handling ambiguous names in the database:

- *Frequent name* = name with frequency >200 in the Patient table.
- *Rare name* = name with frequency <200 in the Patient table.

<table>
<thead>
<tr>
<th></th>
<th>Frequent ambiguous name</th>
<th>Rare ambiguous name</th>
<th>Not ambiguous name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keep patient records in the DB</td>
<td>✔️ e.g. “Hans”</td>
<td></td>
<td>✔️ e.g. “Peter”</td>
</tr>
<tr>
<td>Exclude patient records from the DB</td>
<td></td>
<td>✔️ e.g. “Aaron”</td>
<td></td>
</tr>
<tr>
<td>Replace name in the free text</td>
<td></td>
<td></td>
<td>✔️ e.g. “Peter”</td>
</tr>
<tr>
<td>Do not replace name in the free text</td>
<td>✔️ e.g. “Hans”</td>
<td>✔️ e.g. ”Aaron”</td>
<td></td>
</tr>
</tbody>
</table>
1. The algorithm calculates the frequency of the names.
2. The algorithm replaces names (using the frequency) and numbers with other real names and numbers of the same kind. The new data will look "real".

Replacing by a completely random name, the data pattern might look strange.

E.g. Replacing the common last name “Nielsen” by the rare last name “Pantazos”.
De-identification process took 60 hours:
• 5 hours analyzing and replacing the text and 55 hours updating the tables.

During the de-identification process the system deleted ¼ of the data, 114,315 patient records:
1. Danish ambiguous names: 1,282.
3. Corrupted data and age > 90 years: 69,914.

Without the special rule (frequency >200) we would have lost another 55,000 patients from ambiguous and eponymous names.

The final de-identified EHR database contains 323,122 patient records.
Evaluations of our de-identification algorithm:
A sample of 369 free-text randomly selected from “MedicalRecordLine” table for manual inspection.

<table>
<thead>
<tr>
<th></th>
<th>Should be de-identified</th>
<th>Should not be de-identified</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Was de-identified</strong></td>
<td>a = 1313</td>
<td>b = 109</td>
</tr>
<tr>
<td><strong>Was not de-identified</strong></td>
<td>c = 7*</td>
<td>d = 71,721</td>
</tr>
</tbody>
</table>

*Only one out of 7 was a person name. Frequent ambiguous names were not included.*

<table>
<thead>
<tr>
<th></th>
<th>Formula</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Recall (Hit-rate)</strong></td>
<td>R = [ a / (a + c) ]</td>
<td>99.5%</td>
</tr>
<tr>
<td><strong>Precision</strong></td>
<td>P = [ a / (a + b) ]</td>
<td>92.3%</td>
</tr>
<tr>
<td><strong>F-Measure</strong></td>
<td>[ F = 2 \times (P \times R) / (P + R) ]</td>
<td>95.7%</td>
</tr>
</tbody>
</table>
It is feasible to de-identify an EHR database and achieve an acceptable level of **anonymity, correctness** and **readability** of the medical record.

This database is adequate for:
1. supporting research,
2. software development,
3. training where users are aware of the confidentiality.

It is not adequate for general publication of the database where someone maliciously might look for weaknesses.

The algorithm can be used for other EHRs, but modifications caused by database structure and language should be considered.
Thank you for your attention!
The algorithm consistently replaces person names and CPR-numbers.

- **Person names:**
  E.g. Carla Jensen and John Jensen => Maria Sorensen and Peter Sorensen.

- **CPR-numbers (DDMMYY-CSSG).**
  - **DD** (day) and **MM** (month) are changed to a random, consistent day and month.
  - **C** (century) is not changed.
  - **SS** (serial number) is randomized.
  - **G** (gender) is not altered.
  E.g. number 280210-1546 is replaced with 200610-1656.
Results

De-identification process took 60 hours:
• 5 hours analyzing and replacing the text and 55 hours updating the tables.

Medical records contain a considerable amount of personal identifiers (4%).

<table>
<thead>
<tr>
<th>“MedicalRecordLine” table</th>
<th>Nr. of times</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifiers</td>
<td>13,788,288</td>
<td>4%</td>
</tr>
<tr>
<td>Non-identifiers</td>
<td>322,734,954</td>
<td>96%</td>
</tr>
<tr>
<td>Total</td>
<td>336,523,242</td>
<td>100%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifiers in “MedicalRecordLine” table</th>
<th>Nr. of times</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPR-numbers</td>
<td>455,946</td>
<td>3%</td>
</tr>
<tr>
<td>First names</td>
<td>4,331,593</td>
<td>31%</td>
</tr>
<tr>
<td>Last names</td>
<td>2,675,386</td>
<td>19%</td>
</tr>
<tr>
<td>Emails</td>
<td>18,858</td>
<td>0%</td>
</tr>
<tr>
<td>Phone numbers</td>
<td>43,051</td>
<td>0%</td>
</tr>
<tr>
<td>Streets</td>
<td>3,156,356</td>
<td>23%</td>
</tr>
<tr>
<td>Cities</td>
<td>994,125</td>
<td>7%</td>
</tr>
<tr>
<td>Zip-codes</td>
<td>599,566</td>
<td>4%</td>
</tr>
<tr>
<td>Hospitals</td>
<td>787,055</td>
<td>6%</td>
</tr>
<tr>
<td>Clinics</td>
<td>114,318</td>
<td>1%</td>
</tr>
<tr>
<td>Identifiers</td>
<td>13,788,288</td>
<td>100%</td>
</tr>
</tbody>
</table>
During the de-identification process the system deleted $\frac{1}{4}$ of the data, 114,315 patient records:
1. Danish ambiguous names: 1,282.
3. Corrupted data and age > 90 years: 69,914.

Without the special rule (frequency >200) we would have lost another 55,000 patients from ambiguous and eponymous names.

The final de-identified EHR database contains 323,122 patient records.