Accepted for presentation at MIE2011

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

Kostas PANTAZOS[1], Soren LAUESEN, Soren LIPPERT
Software Development Group, IT-University of Copenhagen

Abstract. Electronic health records (EHR) contain a large amount of structured data and free text. Exploring and sharing clinical data can improve healthcare and facilitate the development of medical software. However, revealing confidential information is against ethical principles and laws. We de-identified a Danish EHR database with 437,164 patients. The goal was to generate a version with real medical records, but related to artificial persons. We developed a de-identification algorithm that uses lists of named entities, simple language analysis, and special rules. Our algorithm consists of 3 steps: collect lists of identifiers from the database and external resources, define a replacement for each identifier, and replace identifiers in structured data and free text. Some patient records could not be safely de-identified, so the de-identified database has 323,122 patient records with an acceptable degree of anonymity, readability and correctness (F-measure of 95%). The algorithm has to be adjusted for each culture, language and database.

Keywords. Electronic Health Record, de-identification, database, confidentiality

1. Introduction

Vast amounts of data are generated from medical systems in structured and free text formats. Although the data exist, clinicians cannot access them due to confidentiality.

The goal of this project is to irreversibly convert patient records from a specific EHR database to unidentifiable records with low distortion of medical correctness and readability. This de-identified database can support research in the healthcare area, improve development of medical software and train new users of the system.

In the medical informatics area, several de-identification algorithms have been developed [1, 4, 5, 6, 7, 8]. Meystre et. al. [3] present a review of recent research on de-identifying electronic health records. Their results showed that most de-identification systems focus on structured data and less on free text. The ones that de-identify free text use mainly predefined medical records (e.g. pathological reports). To our knowledge, previous research focus on de-identifying datasets extracted from tables in an EHR database, and none has presented a de-identification algorithm for a full EHR database, ensuring acceptable levels of anonymity, medical correctness and readability. Furthermore, the literature review [3] showed that previous studies focus more on

[1] Corresponding Author: Kostas Pantazos, IT-University of Copenhagen, Rued Langgaards Vej 7, DK-2300, Copenhagen, E-mail: kopa@itu.dk

anonymity and medical correctness and less on readability of the de-identified records. Finally, this is the first study on de-identifying Danish healthcare records.

2. Challenges

Anonymity can be ensured by finding all the identifiers and altering them. Medical correctness means preserving the medical information as well as ensuring consistency. We defined two types of consistency in an EHR database: internal and external consistency. Internal consistency means that identical identifiers (e.g. civil registration numbers) in the original version are also identical in the new version for each patient. External consistency means that identical identifiers (e.g. last name) in the original version are also identical in the new version across patients. This will for instance preserve family relationships. Readability can be ensured by replacing the identifiers with appropriate real values.

An electronic health record database contains tables with only structured data (e.g. civil registration number and diagnosis name) and tables with free text, often with embedded structured data (e.g. medical notes with a diagnosis name). Preserving anonymity of the patient and medical correctness in structured tables is easy because the context is pre-defined and all identifiers are replaced according to the rules of the format. In contrast, de-identifying free text tables is a challenging task due to the un-defined context, language ambiguities and medical eponyms (e.g. Aaron can be a first name or part of the medical term “Aaron Sign”). Another challenge is to preserve internal and external consistency without affecting medical correctness and anonymity.

3. Solution

We investigated a full 12 gigabyte database with 437,164 patient records containing diagnoses, notes, laboratory data, etc. Figure 1 outlines our process.

3.1.Database Investigation

We examined the database (65 tables) to find tables that might reveal patient identity. We found 9 tables with only structured data and 13 tables with free text. We

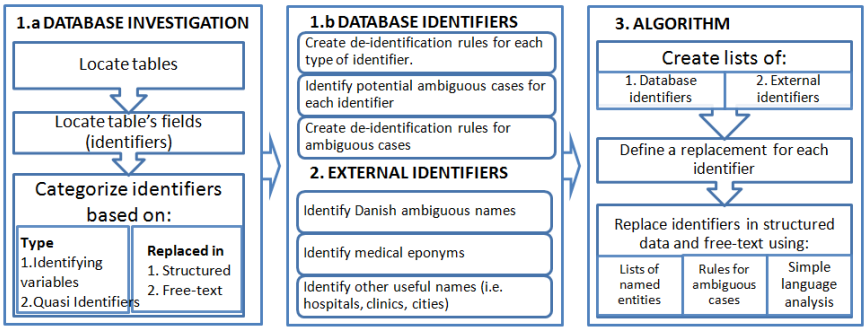

Figure 1. Overview of the de-identification process for an entire Danish EHR database

investigated the fields and created a list of identifiers, e.g. CPR-number (the Danish civil registration number). We also found quasi identifiers (e.g. street name) [2]. In total we found 9 identifiers (CPR-number, first name, middle name, last name, address, telephone number, e-mail, web URL, picture) and 13 quasi identifiers (zip-code, city, country, date of birth, date of death, age, hospital name, clinic name, clinician's first name, clinician's last name, clinician's alias, first name and last name of relatives).

We investigated identifiers and quasi identifiers in the database and found several challenging issues: number ambiguity (a phone number can also be interpreted as a CPR number), language ambiguity (Hans is a Danish pronoun, but can also be a male first name), medical eponymous names (Aaron), city names and clinic names that can also be person names, and corrupted data (invalid CPR numbers in structured data). Our algorithm extracts lists of all the identifiers from the database. The lists are used by the algorithm to identify ambiguous names and numbers in free text.

3.2.External Identifiers

In addition to the identifiers from the structured parts of the database, we used public lists of place names, hospital names, clinic names and medical eponymous names. These names allowed the algorithm to find more ambiguous names in free text, and to de-identify person names that occurred only in free text.

3.3.Algorithm

Structured data: The algorithm replaces all identifiers in structured data. Each family name is consistently replaced by another family name with roughly the same frequency in the database. As an example, the name Nielsen might be replaced by Hansen wherever Nielsen occurs. First male names and first female names are handled in a similar way.

CPR-numbers are consistently replaced by another CPR-number. The CPR format is: DDMMYY-CSSG where DDMMYY is the birth date. The day (DD) and month (MM) are changed to a random, consistent day and month. C stands for century and denotes 1900 or 2000. This is not changed. SS (serial number) is randomized. G shows gender, and is not altered (e.g. number 280210-1546 is replaced with 200610-1656).Some identifiers, e.g. telephone numbers, are replaced by a random number.

Free text: The algorithm looks at each word in the free text and determines whether it is a family name, a male first name, a female first name, a place name, an eponymous medical name, etc. If it is only one of these, it is replaced according to the rule for this kind of name. If it is more than one kind, the word is ambiguous and a special rule is used.

Here is an example of a special rule: If a person name is also an eponymous medical name (Aaron), it should not be replaced. This would destroy medical correctness in case it actually is a medical term. However, if it actually is a person name, keeping the name might harm anonymity. Our special rule is to keep the name if it is a frequent name (occurs more than 200 times). This will have little impact on anonymity. If it is a rare name, we delete the patient entirely from the database.

The algorithm looks at each number and determines by its format and value whether it is a phone number, a CPR-number, etc. If it is only one of these, the corresponding rule is applied. Otherwise the number is ambiguous and the algorithm uses simple language analysis to determine the type.

Original 'Patient' table (structured data)			**De-identified 'Patient' Table**		
CPR-number*	First name*	Last name*	CPR-number	First name	Last name
290210-1546	Carla	Sorensen	200610-1656	Maria	Jensen
Original 'Medical Record Line' table (structured data and free-text)			**De-identified 'Medical Record Line' table (structured data and free-text)**		
CPR-number*	Medical note**		CPR-number	Medical note	
290210-1546	*Copenhagen* Hospital, *Carla Sorensen*, CPR: 290210-1546, Visit date 21-09-2010…"		200610-1656	*Aalborg* Hospital, *Maria Jensen*, CPR: *200610-1656*, Visit date 21-09-2010..."	
290210-1546	*Copenhagen* Hospital, Visit date 22-09-2010...*Carla* visited *Copenhagen* hospital ..."		200610-1656	*Arhus* Hospital, Visit date 22-09-2010...*Maria* visited *Arhus* hospital ..."	

**Structured data, ** Unstructured data(free-text)*

Figure 2. A de-identification example

Consistency: Family doctors often make notes that refer to other family members by name or CPR-number. Since the algorithm consistently replaces person names and CPR-numbers, these references remain consistent. City names, hospital names and clinic names are replaced consistently within a single free text, but not across all free texts. A consistent replacement might expose the identifier since there are rather few replacements for cities, hospitals and clinics.

Readability: Since the algorithm replaces names and numbers with other real names and numbers of the same kind, the new data will look "real". However, if names were consistently replaced by a completely random name, the data pattern might look strange. As an example, the common name Nielsen might be consistently replaced by the rare name Pantazos. As a result we would suddenly have 10,000 Pantazos in the database. For this reason the algorithm replaces a name with a new name of roughly the same frequency.

Figure 2 shows an example of how the algorithm de-identifies data.

4. Results

We evaluated our system manually with a sample of 369 randomly chosen medical free text records extracted from MedicalRecordLine table (7.2 gigabyte). Figure 3 presents the evaluation results. The algorithm did not alter frequent Danish names (>200) that were also medical names. We were aware of this from the beginning but would not

	Should be de-identified	**Should not be de-identified**
Was de-identified	a = 1313	b = 109
Was not de-identified	c = 7*	d = 71,721

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

	Formula	**Value**
Recall (Hit-rate)	R = [a / (a + c)]	99.5%
Precision	P = [a / (a + b)]	92.3%
F-Measure	F = 2 x (P x R) / (P + R)	95.7%

Figure 3. Evaluation results

distort the medical correctness. Since the names are frequent, there is little impact on anonymity. A previous version of the algorithm did not de-identify patient names in genitive form. We adjusted our algorithm to deal with the genitive form. Precision was affected because of the many ambiguous names and abbreviations that were replaced in places where they should not. This had a negative impact on readability and medical correctness. However, the result is very readable because only 109 words out of 71,721 words were wrongly replaced. Anonymity was not affected.

The program took 60 hours using a computer with 4 gigabyte of memory to create the new database (12 days using a computer with 1 gigabyte memory). Of these 60 hours, 5 hours were spent on analyzing and replacing the text and 55 hours on updating the records in the database. During the de-identification process the system deleted ¼ of the data, 114,315 patient records (Danish ambiguous names: 1,282, Medical eponymous names: 43,119, corrupted data and age > 90 years: 69,914). In case we had not used the frequency rule, we would have lost another 55,000 patients from ambiguous and eponymous names. The result of our de-identification process is an EHR database containing 323,122 patient records.

5. Conclusion

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 supporting research, development and training where users are aware of the confidentiality. If you know name, address and CPR-number of a specific person, you will not be able to find his/her health record. However, it is not adequate for general publication of the database where someone maliciously might look for weakness. The principle of the algorithm can be used for other EHRs, but modifications caused by database structure and language should be considered.

References

[1] J. Berman, Concept-Match Medical Data Scrubbing. How Pathology Text Can Be Used In Research, *Archives of Pathology & Laboratory Medicine*, (2003), 680-6.

[2] K. E. Emam, S. Jabbouri, S. Sams, Y. Drouet, Power, M., Evaluating Common De-Identification Heuristics for Personal Health Information. *Journal of Medical Internet Research*, (2006), 8(4):e28.

[3] S. Meystre, F. J. Friedlin, B. R. South, S. Shen, M. Samore, H., Automatic de-identification of textual documents in the electronic health record: a review of recent research. *BMC Medical Research Methodology*, (2010), 10:70.

[4] D. Gupta, M. Saul, J. Gilbertson, Evaluation of a Deidentification (De-Id) Software Engine to Share Pathology Reports and Clinical Documents for Research, *American Journal of Clinical Pathology*, (2004), 176-86.

[5] L. Sweeney, Replacing Personally-Identifying Information in Medical Records, the Scrub System. *In: Cimino, JJ, ed. Proceedings, Journal of the American Medical Informatics Assoc*, (1996), 333-337

[6] G. Szarvas, R. Farkas, R. Busa-Fekete, State-of-the-Art Anonymization of Medical Records Using an Iterative Machine Learning Framework. *Journal of the American Medical Informatics Association*, (2007), 574–8

[7] O. Uzuner, Y. Luo, P. Szolovits, Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, (2007), 550-563.

[8] S. Velupillai, H. Dalianis, M. Hassel, G. H. Nilsson, Developing a standard for de-identifying electronic patient records written in Swedish: Precision, recall and F-measure in a manual and computerized annotation trial, *International Journal of Medical Informatics*, (2009), 78-90.