De-Identification Algorithm for Free-Text Nursing Notes

MM Douglass[1], GD Cliffford[1], A Reisner[1], WJ Long[2],
GB Moody[1], RG Mark[1]

[1]Harvard-MIT Division of Health Sciences & Technology, Cambridge MA, USA
[2]Laboratory for Computer Science, MIT, Cambridge MA, USA

Abstract

All personally identifiable information must be removed from patient medical records before the data can be shared with other researchers. We present an automated method of removing protected health information (PHI) from free-text nursing notes taken from a U.S. hospital. We have previously shown that one clinician can locate PHI in nursing notes with an average sensitivity of 0.81, and for teams of two clinicians the sensitivity is 0.94. Our method uses lexical look-up tables, regular expressions, and simple heuristics to locate PHI with an overall sensitivity of 0.92 (0.98 for names, 0.96 for dates), which is significantly better than the average sensitivity of a single human. The algorithm has a positive predictive value of only 0.44, so additional software was developed to allow the user to review the terms identified as PHI and manually eliminate false positives. The algorithm is open-source and will be made freely available on PhysioNet together with a re-identified corpus of nursing notes.

1. Introduction

Patient hospital records are invaluable resources for biomedical research, but they contain highly sensitive personal information that must be kept confidential. The de-identification process removes all information that can be used to identify who the patient is, while still preserving all the medically relevant information.

Guidelines for protecting the confidentiality of health care information have been established in the United States in the Health Information Portability and Accountability Act (HIPAA) of 1996 [1]. Records are said to be de-identified when the risk is very small that the information can be used alone or in combination with other reasonably available information to identify the individuals. This risk can be calculated and documented statistically for all the records, or we can use the safe harbor approach and show that every record is free of the 18 types of identifiers listed in the law. Those identifiers include: names, all geographic subdivisions smaller than a state, all elements of dates (except year) for dates directly related to an individual, including birth date, admission date, discharge date, date of death; all ages over 89, telephone and fax numbers, social security numbers, and medical record numbers. Such data is known as protected health information (PHI).

Extensive medical data is being collected from patients admitted to the intensive care units of local hospitals as part of the MIMIC II project [2]. The nursing progress notes are unstructured free text typed by the nurses at least twice a day, and include observations about the patient's medical history, his current physical and psychological state, medications being administered, laboratory test results, and other information about the patient's state. In these notes, the nurses frequently employ technical terminology, non-standard abbreviations, ungrammatical statements, misspellings, and incorrect punctuation and capitalization. A sample note is given in Figure 1.

Applying traditional natural language processing techniques to the biomedical domain is difficult because of the lack of relevant training data necessary for

> 79 YO FEMALE BROUGHT TO GH CATH LAB TODAY FROM OSH FOR ELECTIVE CARDIAC CATHERIZATION. PT ADMITTED TO OSH 8/2 WITH C/O CHEST PAIN. R/O'D MI. REPORTED TO HAVE +ETT AND SENT FOR CATH. PT FOUND TO HAVE 70% LM OCCLUSION, 60% LCX OCCLUSION, AND 80% RCA PROXIMAL. STENTS PLACED TO LCX AND LM. ONCE IN HOLDING ROOM PT VOMITED APPROX 400CC BRIGHT RED BLOOD WITH SIGNIFICANT DECREASED IN BP. IV HEPARIN, NTG, AND INTEGRELLIN DC'D AT THAT TIME AND PT GIVEN IVF. ADMITTED TO CCU FOR CLOSER MONITORING.
>
> PMH: S/P MI 10/98, S/P CABG 10/98 , EF 30%, GLOBAL HK , HTN, ^CHOL

Figure 1: Sample nursing note.

statistical methods and the specialized terminology and frequent use of ambiguous abbreviations that confound rule-based methods. Techniques for de-identifying medical text have been developed by other groups ([3-5]), but none of these techniques is designed to be used on data as ungrammatical and unstructured as our nursing note corpus.

2. Methods

The algorithm requires the installation of Perl (version 5.8.1 and above) and its String::Approx module. The intended input is a single file containing free-text nursing notes. The output is the nursing notes with dates shifted according to a random offset and all the other PHI replaced with appropriate fake data. The algorithm does not depend on the availability of any outside information about the patients or the dates of treatment.

An overview of the de-identification algorithm is shown in Figure 2. For each term in the note text, the algorithm first determines whether it contains numbers. Regular expressions are used to determine whether numeric tokens should be classified as dates, telephone numbers, or other types of identifying numbers. Non-numeric tokens are classified using lexical matching and by applying simple heuristics, as explained in the following sections.

2.1. Finding Names

The most important type of data we need to remove with 100% accuracy is the patient's name. A single mention of the patient's name in publicly released data would be an unacceptable violation of privacy. We could obtain the patient's name from certain tables in the MIMIC II database and then search for and remove occurrences found in the nursing notes. However, in the nursing note the name could be spelled incorrectly (``Willaim'') or the patient may use a nickname (``Bill''), so the algorithm cannot rely on being provided with the name information. We also want to identify and remove the names of the other people mentioned in the notes, including visiting relatives and the attending clinicians.

The algorithm initially identifies occurrences of common first names by lexical matching of words from the nursing notes with all names in the lists of female and male first names obtained from the 1990 U.S. Census [6]. Next, it looks for misspellings of the 100 most popular male and female first names using Perl's approximate matching module String::Approx. The potential first names are classified as "ambiguous" and "unambiguous" names based on whether the names are also found on a list of standard English words obtained from the Spell Checking Oriented Word Lists [7] or on the list of

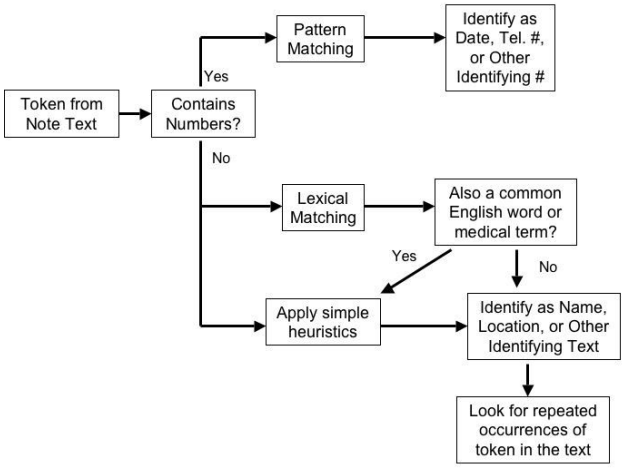

Figure 2: Overview of the de-identification method.

medical terms from the Unified Medical Language System (UMLS) [8]. If a name is labeled "unambiguous", every occurrence of it will be removed from the text. If a name is "ambiguous", simple heuristics are used to determine whether to remove it. First names are usually found before a last name or close to a word like ``wife", ``friend", or ``nurse", that identifies who the person is. Our method looks at the context of "ambiguous" names to locate potential last names and identifiers that would indicate that the "ambiguous" name should be removed.

Last names are identified by heuristics and by lexical matching with a portion of the list of last names from the 1990 U.S. Census. In nursing note texts, the last names are always preceded by a first name, the individual's initials, or a title. Not all words that look like titles may be functioning as titles. For example, "MS" could be a title, or it could stand for milliseconds or multiple sclerosis. Single letters followed by a "." are not always initials, and not all initials are followed by ".". The heuristics must be very flexible to be used with the ungrammatical nursing note text, but because of the rules' flexibility many words are incorrectly labeled and removed as names.

2.2. Finding Locations

The names of locations smaller than a state are found in names of hospitals, where the patient comes from, and where the patient's visitors are from. The algorithm locates occurrences of hospital names by using a list of all the local hospitals and by looking for misspellings of hospital names. Since most patients will be coming from the area around the hospital, the algorithm uses lists of towns and cities in the area to locate the names of local places. The patients' visitors can come from anywhere

around the world, so the algorithm uses lists of major cities in the US and the world, and it uses simple heuristics to try to pick out cities that are not on the lists or that are misspelled.

2.3. Finding Dates

Dates are written in many different ways in the nursing notes ("5/22/99", "5-22", "May 22 1999", "May 22nd", "the 22nd"). The algorithm looks for patterns of numbers that look like dates, and it specifically looks for the month names and then looks for the days and years around the month.

A year by itself often appears in the patient medical history (ex. ``cholecystectomy, 1953''). We tried many different methods of locating isolated years, but none worked well. In the end, we decided to allow the years to remain. HIPAA does not require the removal of years unless they are indicative of an age above 89 [1]. Our nursing notes never mention date of birth, so we can safely leave in all years after 1915. (Of course, this date must be incremented yearly.) The major disadvantage in not being able to locate and remove all the years is that we will be unable to automatically shift those years to correspond to the time shift in the other dates in the notes.

2.4. Finding Other PHI

The algorithm uses a look-up list to identify building names, specially named wards, and any other words or phrases that would identify which hospital the patient is in. Telephone numbers are found by matching regular expressions or by looking for long strings of numbers that are preceded or followed by "telephone", "pager", "mobile", or other related terms. Other types of identifiers, like social security numbers, are identified by looking for potential indicators like "SS" or "Id" followed or preceded by a series of digits not otherwise labeled.

2.5. Repeated Occurrences of PHI

The same names often reappear in the notes for a single patient. The patient's son may visit often, or the same clinicians may see the patient during her stay. The algorithm looks for repeated PHI instances within the collection of notes for a single patient.

Initially all the non-numeric PHI instances – the names, locations, and hospital names – are collected from all the notes for the patient. Then the algorithm compares the list of unique PHI instances with a list of common English words (from the Spell Checking Oriented Word Lists at size 10 [7]). PHI instances that are on the list of common words are removed from the list. The resulting list is used by the algorithm to identify other occurrences of already found PHI in the patients' notes.

2.6. Elimination of False Positives

We realized from preliminary tests that our automated methods have high false positive rates, and we found in a previous study that human manual de-identification has very low false positive rates [9]. Since humans can easily and quickly distinguish false positives from true positives, we created software that displays all the choices made by the algorithm and allows the user to decide whether the identified PHI should be removed. The software shows the user the location of the term within the original note so she can look at the context when deciding whether the term is PHI. The output of the software is used to create the final de-identified version of the notes.

3. Results

We have previously created a large gold standard collection of manually de-identified nursing notes [9]. Three clinicians independently identified PHI in every note, and the selections were combined and reviewed by a fourth clinician, who adjudicated whether the identified PHI had been labelled correctly. As a final test, a simple algorithm developed for in-house use was ran on the notes to identify additional PHI. We used a portion of that database for testing our de-identification method.

The algorithm was tested on 747 nursing notes taken from 22 patients, containing 99,443 words with 411 instances of PHI. The results are shown in Table 1.

A major source of false positives came from the part of the algorithm that looked for repeated occurrences of already found PHI. Because of the high false positive rate when identifying potential names, many common terms are currently being tagged as PHI, and then the code looks for every other occurrence of the word in the other notes for that patient. The algorithm currently checks to see whether the found PHI is a commonly used word, but the reference lists of common words are based on which words are the most commonly used in non-medical, correctly spelled, grammatically correct English texts. None of the common nursing terminology or

Type	TP	FP	FN	Sens	PPV
Name	139	178	3	0.98	0.44
Date	160	132	6	0.96	0.55
Overall	378	490	33	0.92	0.44

Table 1: Results for the algorithm on a collection of 747 nursing notes, containing 99,443 words with 411 instances of PHI. TP = True Positive, FP = False Positive, FN = False Negative. Sens = Sensitivity, PPV = Positive Predictive Value.

abbreviations are found in those lists. Errors related to repeated occurrences of incorrectly identified PHI account for 148 of the false positives. This part of the code was not removed because it does find missed names, locations, and other correctly identified PHI.

The false negatives were most frequently names of other local hospitals that were either not included in the list of hospitals used by the algorithm or the names were abbreviated in ways not included in the list.

As mentioned earlier, the most important type of PHI to identify is names. The algorithm's ability to locate names is very good: only three out of 142 names were missed in all the notes tested. Two of those names were of hospital employees. The algorithm can be easily altered to use the list of all the hospital employee names in MIMIC II. The other undetected name was a misspelled name that contained too many errors for it to be found with Perl's approximate matching function.

4. Discussion and conclusions

In our earlier study of human de-identification, we found that a single person can recognize PHI with an average sensitivity of 0.81 and positive predictive value of 0.98, and the union of two people find PHI with an improved sensitivity of 0.94 and PPV of 0.97 [9]. Even in its current imperfect form, the algorithm's performance is better than the average single person and is nearly as good as two people de-identifying the text. (The performance statistics cannot be compared exactly because the algorithm does not look for single years found alone. The human de-identifiers were looking for more types of PHI than the algorithm currently identifies.)

The tests we have performed have suggested simple rules that we should add to the algorithm, and the tests have exposed the shortcomings in our look-up tables, such as the lack of common abbreviations and drug names. Having lists of the common words and abbreviations found in hospital nursing notes would help reduce the false positive rate. More work must be done on the algorithm so we can eventually have a fully automated method that reliably performs better than human de-identifiers without requiring the extra step of manually reviewing the selections and removing the false positives.

One likely future avenue in our work is the use of Hidden Markov Models to incorporate contextual information learned from a much more massive corpus of data. Such techniques may obviate the need to trawl almost endless lists of PHI and compile rules of exceptions, which may conflict.

We have developed an automated method for de-identifying free-text nursing notes that allows us to remove PHI from patient records with a high sensitivity. The algorithm is open-source and will be made freely available on PhysioNet [10, 11] together with a re-identified corpus of nursing notes.

De-identification is an important and necessary process when using hospital patient data, and the better our de-identification methods are, the faster we can obtain more data and make it available to the biomedical research community.

Acknowledgements

This publication was made possible by Grant Number R01 EB001659 from the National Institute of Biomedical Imaging and Bioengineering.

References

[1] Health Insurance Portability and Accountability Act of 1996.

[2] Saeed M, Lieu C, Raber G, Mark RG. MIMIC II: A massive temporal ICU patient database to support research in intelligent patient monitoring. Computers in Cardiology, 29:641–644, 2002.

[3] Sweeney L. Replacing personally-identifying information in medical records, the scrub system. Proc AMIA Symp, pages 333–337, 1996.

[4] Ruch P, Baud R, Rassinoux A-M, Bouillon P, Robert G. Medical document anonymization with a semantic lexicon. Proc AMIA Symp, pages 729–733, 2000.

[5] Gupta D, Saul M, Gilbertson J. Evaluation of a de-identification software engine: Progress towards sharing clinical documents and pathology reports. Am J Clin Pathol, 121(2):176–186, 2004.

[6] U.S. Census Bureau. 1990 census name files, 1999. http://www.census.gov/genealogy/names/.

[7] Atkinson K. Spell checking oriented word lists, revision 6, 2004. http://prdownloads.sourceforge.net/wordlist/scowl-6.tar.gz.

[8] Lindberg DA, Humphreys BL, McCray AT. The unified medical language system. Methods Inf Med, 32(4):281–91, Aug 1993.

[9] Douglass M, Clifford GD, Reisner A, Moody GB, Mark RG. Computer-Assisted De-Identification of the Free Text in the MIMIC II Database. Computers in Cardiology 2004.

[10] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PC, Mark RG, Mietus JE, Moody GB, Peng CK, Stanley HE. Physiobank, Physiotoolkit, and Physionet: Components of a new research resource for complex physiologic signals. Circulations 2000;101(23):e215–e220.

[11] http://www.physionet.org/.

Address for correspondence:

Margaret Douglass
Laboratory for Computational Physiology
Harvard-MIT Division of Health Sciences & Technology
Rm E25-505, 45 Carleton St.,
Cambridge MA 02142 USA
douglass@mit.edu