Building a Best-in-Class De-identification Tool for Electronic Medical Records Through Ensemble Learning

Karthik Murugadoss[1], Ajit Rajasekharan[1], Bradley Malin PhD[2], Vineet Agarwal[1], Sairam Bade[1], Jeff R. Anderson PhD[3], Jason L. Ross[1], William A. Faubion Jr., MD[3], John D. Halamka, MD[3], Venky Soundararajan PhD[1], Sankar Ardhanari[1*]*

[1]nference, Cambridge MA, USA
[2] Vanderbilt University Medical Center, Nashville TN, USA
[3] Mayo Clinic, Rochester MN, USA
*Address Correspondence to Venky Soundararajan (venky@nference.net), Sankar Ardhanari (sankar@nference.net)

Abstract

The natural language portions of an electronic health record (EHR) communicate critical information about disease and treatment progression. However, the presence of personally identifying information in this data constrains its broad reuse. In the United States, the Health Insurance Portability and Accountability Act of 1996 (HIPAA) provides a de-identification standard for the removal of protected health information (PHI). Despite continuous improvements in methods for the automated detection of PHI over time, the residual identifiers in clinical notes continue to pose significant challenges - often requiring manual validation and correction that is not scalable to generate the amount of data needed for modern machine learning tools. In this paper, we describe an automated de-identification system that employs an ensemble architecture, incorporating attention-based deep learning models and rule based methods, supported by heuristics for detecting PHI in EHR data. Upon detection of PHI, the system transforms these detected identifiers into plausible, though fictional, surrogates to further obfuscate any leaked identifier. We evaluated the system with a publicly available dataset of 515 notes from the I2B2 2014 de-identification challenge and a dataset of 10,000 notes from the Mayo Clinic. We compared our approach with other existing tools considered best-in-class. The results indicated a recall of 0.992 and 0.994 and a precision of 0.979 and 0.967 on the I2B2 and the Mayo Clinic data, respectively.

Introduction

The widespread adoption of electronic health records (EHRs) by healthcare systems has enabled digitization of patient health journeys. While the structured elements of EHRs (e.g., health insurance billing codes) have been relied upon to support the business of healthcare and front office applications for decades, the unstructured text (e.g., history & physical notes and pathology reports) contains far richer and nuanced information about patient care, supporting novel research [1-5]. However, this text often contains protected health information (PHI) as defined in the Health Insurance Portability and Accountability Act of 1996 (HIPAA), such as the personal name, phone number, or residential address [6]. As a consequence, such data has limited reuse for secondary purposes [7].

HIPAA permits data derived from EHRs to be widely shared when it is de-identified. Under the HIPAA Privacy Rule, de-identification can be accomplished in several ways. The most straightforward is the Safe Harbor implementation, which necessitates removal of an enumerated list of 18 categories of direct- (e.g., Social Security Number) and quasi-identifiers (e.g., date of service).

The notion of de-identification for natural clinical language is not new [8, 9]; however implementing a scalable approach has been challenging due to several competing requirements. First, from a regulatory perspective, it must achieve extremely high recall, in that it needs to detect nearly, if not, all instances of PHI. Second, from a clinical utility perspective, it must achieve extremely high precision, so that we maximize the correctness of biomedical research performed. And, third, the approach needs to be cost effective, so that millions of records can be de-identified in a reasonable amount of time. The traditional approach of manual detection of PHI is expensive, time consuming and prone to human error [10-13], which makes automated de-identification a more promising alternative. However the competing requirements of high recall and high precision has been difficult to achieve through automated approaches to date [10]. In this paper, we integrate a collection of approaches, blending the beneficial aspects of modern deep learning along with rules and heuristics, to create a best-in-class approach to automated de-identification. To mitigate risk for any residual PHI, the system transforms each detected PHI instance into a suitable surrogate **(Fig. 1)**.

This paper begins with a review of the key characteristics that distinguish state-of-the-art machine learning models from more traditional models. Next, we present the ensemble architecture and the methods employed to overcome deficiencies in the state-of-the-art. Then, we examine the performance of this architecture for detecting PHI elements in two datasets, the publicly accessible I2B2 2014 de-identification challenge dataset and a substantially larger and diverse dataset from the Mayo Clinic. We close this paper by highlighting opportunities to refine the ensemble approach and ensure its generalizability for reuse in new settings.

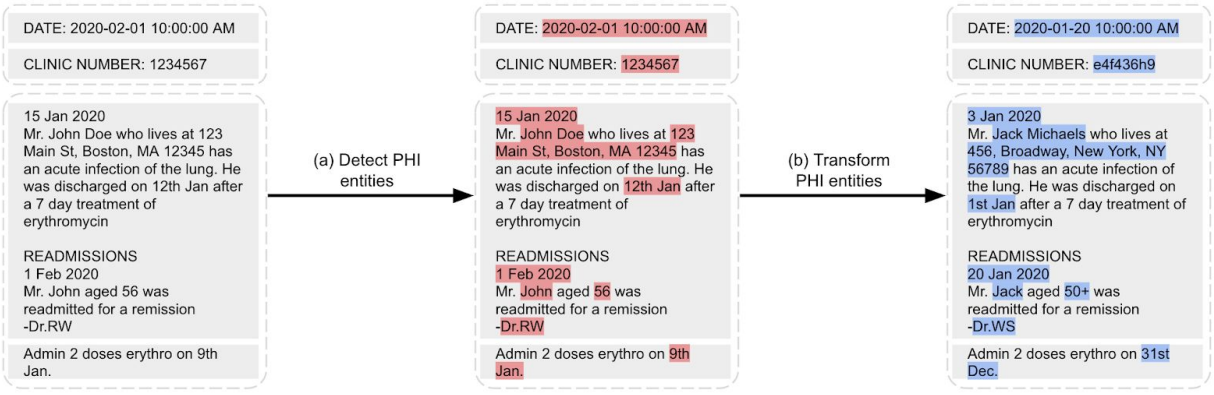


Fig. 1 *Automated de-identification of EHRs involves two steps: (a) Detecting PHI entities and (b) Transforming them by replacement with suitable surrogates.*

Background

Several recent advancements in natural language processing (NLP) have created an opportunity to simultaneously achieve high recall and high precision in automated PHI de-identification. First, unsupervised learning from a language modeling task (e.g., autoregressive and autoencoder models [14]) on a large corpus can now be transferred with very little additional data to a supervised task such as named entity recognition (NER). Second, attention-based deep learning models, such as transformers, allow for the non-sequential processing of text [15]. These methods augment vector representations for tokens (words or subwords) with positional encoding and incorporate a self-attention mechanism, which dynamically computes the weighted sum of the relationships between tokens and its neighbors. Each token is subsequently transformed into a richer context-aware representation. Taken together, positional encodings and the attention mechanism allow transformers to circumvent the long term dependency problem faced by recurrent neural networks (RNNs) and long short term memory networks (LSTMs), leading to better performance on a wide range of NLP tasks while being computationally less expensive than their predecessors. Third, semantic segmentation algorithms generate a fixed vocabulary size constructed with subwords (which also includes single characters) [16-18]. This has the benefit of eliminating out-of-vocabulary words as well as bounding memory requirements, even for very large corpora. Finally, the traditional transformer architecture has been improved upon through bidirectional encoder representations from transformers (BERT) [19] and similar technologies that jointly train a *masked language model* (MLM) pre-training objective and a *next sentence prediction* task.

These breakthroughs set the stage for learning context independent representations of terms in text, and training context-sensitive models that transform those representations into context-aware representations based on the occurrence of a term in a sentence. The de-identification system we report in this paper leverages these improvements in the downstream task of NER of PHI in text.

Overview of de-identification systems

Automated de-identification systems can broadly be segmented into four categories.

Rule-based systems [20-22, 40, 41] use pattern matching rules, regular expressions, dictionary and public database lookups to identify PHI elements. These are simple to implement and usually deterministic; however, these systems have several drawbacks. First, pattern matching rules for identifiers are typically not robust for handling variance in input due to typographical errors(spelling, punctuation, casing etc.); A rule that matches *"Dr. John"* may not be able to match *"Dr john"*. Second, creating template patterns to match sentence fragments like *"Provider Name: Dr. John"* that tag any term after *"Provider Name: Dr."* as a name, for example, requires manual effort to understand the data to create these templates. Doing this for large data sets with notes for millions of patients is time consuming and intractable. Third, dictionary-based systems may not be complete, resulting in increased 'false negatives' (i.e. true PHI that is not detected). Fourth, blindly using dictionary/database lookups induces 'false positives' because they tag phrases that are not identifiers in the context in which they are used

that need to be disambiguated [23] . For example, in *"The doctor determined his Braden Score as normal"*, the term "Braden" might be flagged as PHI, when it is only a clinical term.

Traditional Machine Learning (ML) systems [24-26, 42] use traditional machine learning (ML) algorithms, such as support vector machines (SVMs) and conditional random fields (CRFs), to perform NER classification as PHI for each word in a sentence. The classification task involves creating labeled data and defining features based on properties like part of speech (POS) tags, typography (e.g., capitalization, casing, spacing, font weights, or font types), punctuation, and frequency of words and/or their neighbors. These methods, in addition to requiring significant effort in encoding the feature vectors, may not generalize across datasets.

Deep Learning systems [27, 28], have become the state-of-the-art for a wide variety of application domains, including vision (e.g., image classification) and speech (e.g., voice recognition and generation). In language-related tasks (e.g machine translation), these approaches have surpassed human level performance [29]. Deep learning has proven beneficial in numerous NLP tasks, including predicting the next word (language modeling), tagging tasks such as part of speech tags, entities in a sentence (entity recognition), and dependency parsing. This has enabled applications that traditionally required custom rules and hand-crafted features to be solved without any feature engineering.

Modern deep learning approaches for de-identification have been shown to outperform their predecessors [18], but they require very large quantities of domain specific labeled training data to perform well. Specifically, the challenges include, but are not limited to, the presence of long and highly descriptive sentences, usage of clinical shorthand (that vary across physicians and medical specialties), and a variety of semi-structured machine generated content. Moreover, publicly available datasets for de-identification (including the popular i2b2 2014 dataset [30]) lack diversity, often focusing on only a few types of notes or areas of disease. Training and benchmarking with such datasets is likely to bias the resulting models and fail to capture the nuanced and complex nature of physician notes.

Hybrid [31] and Ensemble Systems [**32-34**] use combinations of rule-based and machine learning-based components in tandem to improve PHI detection efficacy. With these approaches, the choice of components, finding the right split of tasks between them and the optimal strategy for combining results from them become crucial. Some approaches [35, 36] invoke engineering post-processing layers that fix the errors that are introduced by other (earlier) components. In cases where there is, by design, overlap in the type of PHI being predicted (e.g. multiple components detecting people names), considerable effort is spent measuring and choosing a method, like a stacked meta classifier or voting scheme, to pick a winning component [32].

Methods

This section describes how PHI is detected and subsequently transformed into suitable surrogates through our system.

Detection of PHI entities

The ensemble architecture described in this section leverages state of the art attention-based deep learning models (BERT, XLNet, and variations of these models) [14, 19] in conjunction with rules harvested from the data (as described below) to handle semi-structured text. **(Fig. 2)**

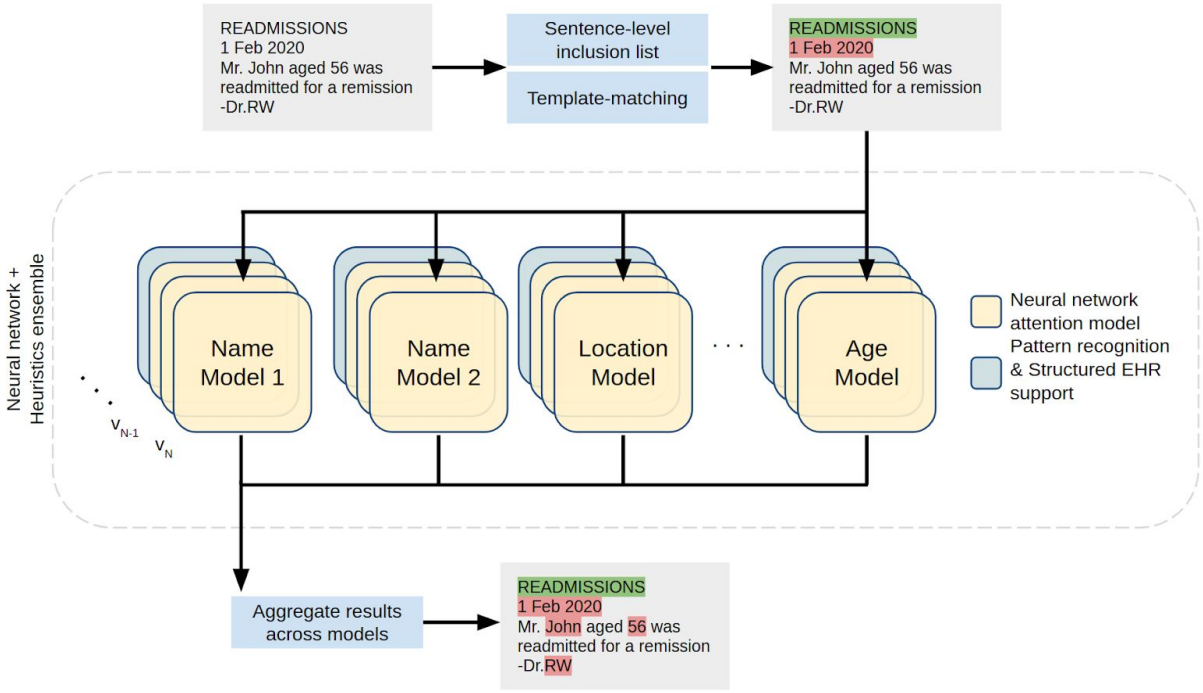


Fig. 2 *Sentence-based inclusion lists and template matching prune out sentences that either 1) lack PHI or 2) contain PHI in specific well-defined patterns. An ensemble of attention-based neural networks identify complementary features across different PHI types. For each entity type, multiple model versions (v_1, v_2, … , v_N) are used in tandem. Additionally, pattern recognition modules and structured EHR content from matched patients support the anonymization process. The results from each component of the ensemble are aggregated to yield the original note labelled with PHI tags.*

There are several salient features of this approach that are worth noting.

Hybrid Deep Learning Models: The newer breed of attention based deep learning models, in conjunction with transfer learning, allow for faster tuning of these models with significantly smaller sets of labeled data for detecting PHI identifiers. We use pre-trained language models that are then fine tuned for detecting (a) personal names, (b) organizations, (c) locations, and (d) ages. The fine-tuning process involves training the pre-trained language model on a named entity recognition task using a training set of annotated sentences. The model is then evaluated on a validation dataset and accuracy scores were computed. The fine-tuning and model validation process are performed in an iterative manner (see Supplementary Methods and **Supplementary Table 1** for details). These types of identifiers are well suited to a statistical entity recognition method because they are able to use the context of surrounding text to disambiguate the entity type of a word. By contrast, pattern matching rules are significantly hampered in this respect. For instance, to detect “Glasgow” as a medical term in *“He had no*

helmet and his Glasgow Score was 6" and as a location in *"Mr. Smith had visited his family in Glasgow"*.

However, we use patterns to deterministically tag reasonably well-defined PHI identifiers, which are almost entirely context independent and unambiguous. This category includes dates and times, phone and pager numbers, clinical IDs and numeric identifiers, email, URLs, IP addresses, and vehicle numbers. In addition, harvested sentence templates (described further below) are relied upon to deterministically tag PHI instances matched by the template patterns. Our methods apply to any such content in both structured(e.g lab comments) and free form text(e.g progress notes).

Additionally, it should be noted that we designed our method to enable organizations to detect and transform information about those who provide care, such as physicians, nurses, and pharmacies. Though this is not required by HIPAA Safe Harbor, this allows organizations to protect the identities of their employees as well.

Ensemble of models framework and iterative fine tuning: Given the regulatory necessity of extremely high recall for de-identification, we run multiple models trained for the same PHI type. In this respect, if a term is detected as PHI in any of the models for that type, then it is tagged. A divide and conquer approach has been implemented that harnesses the power of multiple models to identify PHI or extract meaningful entities (***Fig. 2***). In contrast to a "one size fits all" model, this framework allows each individual model to be fine-tuned to learn different complementary features of the unstructured EHR data as has been shown to be used in previous systems[37]. For instance, one model focuses on identifying peoples' names while another is geared towards addresses and locations.

Furthermore, there are additional models corresponding to cased and uncased variants of the raw data (referred to as "*Name Model 1*" and "*Name Model 2*" in **Fig. 2**). Each model here corresponds to an attention-based deep neural network. One advantage of carving out the entity space to be handled individually by separate models is that each model needs to only learn the distribution of entities of a specific type as opposed to all entities. However, this introduces a challenge in resolving terms in a sentence that have conflicting and/or ambiguous entity types. These conflicts are resolved in the aggregation phase of our ensemble where a simple voting threshold of one claim is employed (i.e., an entity is considered PHI even if one model in the system tags it as such). Since the majority of the components in the ensemble are designed to detect complementary features, we are able to improve recall without much loss of precision.

Integrating databases as part of core model: We use publicly available databases of names, locations, and addresses to augment model fine-tuning. This enables information from these databases to inform model decision making directly. In addition, patient-specific information from structured EHRs, including patient names and residential addresses, are used to augment the model training and match against PHI in the text.

Sentence-based inclusion list: Clinical note corpora contain a large number of repeated sentences. These stem from various processes, including automated reminders (e.g., "*Please let your doctor know if you have problems taking your medications*"), repeated phrases in the writing style of physicians (e.g. "*Rubella: Yes*", "*Pain symptoms: No*") or shared elements in the clinical notes such as section headers (e.g. "*History of Present Illness*"). From the corpus of physician notes from the Mayo Clinic, a set of 1,600 sentences, that did not contain PHI, were incorporated into an "inclusion list". This inclusion list was further expanded with a set of 25,000 sentences containing medically relevant entities, such as disease or drug names (see

Supplementary Methods for details on how the inclusion list was constructed). This has the added benefit of improving the precision of the de-identification system because it reduces the risk of misclassifying these important entities as PHI by the neural network models. Additionally, sentences marked as being devoid of PHI during the validation phase in the iterative fine-tuning process are also added to the inclusion list (see Supplementary Methods).

Auto-Generating templates using statistical NER models: In addition to exact sentences with high prevalence there are also a large number of PHI containing sentences that can be mapped to a template (e.g., *"Electronically signed by: SMITH, JOHN C on 01/02/1980 at 12:12 PM CST"* maps to a template of the form "*Electronically signed by: <LAST NAME>, <FIRST NAME> <INITIAL> on <DATE> at <TIME>"*). While machine learning NER models can be trained and/or fine tuned to learn these patterns, there are instances where entity recognition fails. So, though a name of the form "SMITH, JOHN C" might be detected, "DEWEY" in "DEWEY, JONES K" may not be detected. By contrast, regular expression rules faithfully match every PHI for these cases.

The problem, however, is that the process of identifying such templates and generating the corresponding regular expressions is arduous because it involves manual inspection of a sufficiently large sample of sentences in the corpus. Thus, we use the PHI detecting NER ensemble models to aid in the harvesting of these pattern templates. Every sentence from a large enough sample to represent the corpus is passed through the ensemble and detected PHI is transformed to its corresponding IOB2 mask (e.g., *"Electronically signed by: B-PER I-PER I-PER on B-DATE at B-TIME PM CST"*) generating a potential NER template. Additionally, a 'syntax template' for these sentences is also generated, such that any term that was detected as an entity is mapped to its syntactic representation - one of 'W' for alphabets only, 'N' for numbers only and 'A' for alphanumeric (e.g., *"Electronically signed by: W, W W on N/N/N at N:N PM CST"*). Finally, for each unique syntax template, if there exists only one NER template amongst all instances of the syntax template, a regular expression rule is generated (e.g *"Electronically signed by: [A-Za-z]+, [A-Za-z]+ [A-Za-z]+ on \d+/\d+/\d+ at \d+:\d+ PM CST"*) by mapping each syntax token to its corresponding regular expression pattern - 'W' to '[A-Za-z]+', 'N' to '\d+' and 'A' to '\w'.

Transformation of tagged PHI entities

The de-identification process is designed to recognize words and phrases that represent PHI and other sensitive elements with high recall. However, if the input text is transformed to the de-identified version by *redacting* detected PHI, undetected PHI (e.g., 'Hayley' and the date '7/21' in **Fig. 3**) is obviously leaked to any person who reads the document. As such, the obfuscation process aims to conceal these residual PHI by *replacing* detected PHI with suitable surrogates so it is difficult to distinguish between the residual PHI and the surrogates. As highlighted in **Fig. 3**, from the output of the replacement strategy, it is difficult for a human to determine if either "Jack Michaels" or "Hayley" is a leaked instance of PHI. Evidence with human readers, using this mechanism of *Hiding in Plain Sight (HIPS)* [38], has shown that when the recall of a natural language processing tool is high (i.e., when most real identifiers are detected), the rate of distinguishing real from filler identifiers is no better than what one would encounter by random chance. It has further been shown, however, that under highly controlled conditions, it is possible for a machine learning system to replicate the behavior of the natural language de-identification tool to remove fillers and leave real identifiers in place [39].

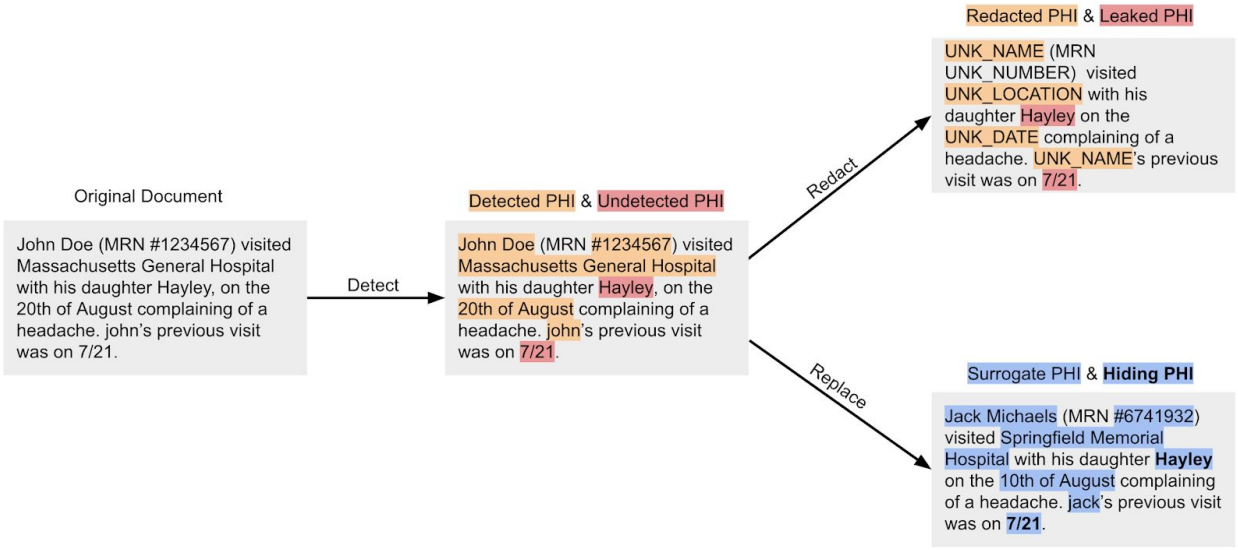


Fig. 3 *An illustration of the hiding in plain sight (HIPS) mechanism to highlight the utility of the detect → replace strategy. After obfuscation, distinguishing real PHI from surrogates is no better than what one would expect by random chance.*

In addition to employing the HIPS method, we apply entity specific rules and heuristics to improve the fidelity of the surrogate. We further improve interpretability of the output by ensuring that every unique PHI token in all EHR records for a patient has the same transformation. Consider the input text "*John Smith, a pleasant 67 year old presented with his son Jack. John complains of breathing difficulty*" was transformed to "*Jane Kate, a pleasant 67 year old presented with his son Matt. Ryan complains of breathing difficulty*." In this example, "*Jane Kate*" as a surrogate is an obvious giveaway that it is a fake name and therefore lends itself to be distinguished from any true PHI that may have leaked. Furthermore, it appears that a third completely different person is complaining of breathing difficulty. So an ideal transformation would have maintained the format of first name followed by last name and the gender for "John Smith" and every instance of "*John*" or "*Smith*" in the input would be transformed to the same output; something like "*Jacob Hamilton, a pleasant 67 year old presented with his son David. Jacob complains of breathing difficulty*."

As discussed, we handle the replacement of surrogates per entity type (see **Supplementary Table 2**). Names are transformed in a manner that is consistent with format, gender and ethnicity of the original (i.e., "*Ms. Lopez visited New York General Hospital for her routine checkup"* becomes "*Ms. Hernandez visited Mass General Hospital for her routine checkup"*). Dates are handled in a way to preserve their formatting (i.e., "*March 5th, 2014*" becomes "*February 27th, 2014*" and "*03-05-2014*" becomes "*02-27-2014*"). The shift in the date is a patient-specific random number. This ensures that dates are shifted consistently for a given patient. Locations and organizations are replaced with suitable surrogates chosen from a predefined dictionary. PHI entities that contain numeric digits (such as phone number or patient ID) involve replacing these numbers randomly while maintaining overall length and format.

While the transformation output of an input token is the same for all instances of its occurrence for a given patient, they would be different across patients. That is, while all instances of “John” in one patient might be transformed to “Jacob” for another patient it could be “Aaron”.

Dataset Description

I2B2 Dataset Description

The I2B2 2014 De-identification and Heart Disease Risk Factors challenge [29] is a publicly available dataset of clinical documents with annotated PHI elements. This dataset consists of a training set of 792 clinical notes and a test set of 515 clinical notes. We used this validation set to benchmark the performance of our approach against existing de-identification methods.

Mayo Dataset Description

The Mayo EHR dataset is based on data from 477,000 patients that originated from multiple EHR data systems (including Epic and Cerner) spanning over 20 years. The dataset included 104 million physician notes that capture the healthcare journey of patients in addition to structured tables containing lab test measurements, diagnosis information, orders, and medicine administration records. This research was conducted with approval from the Mayo Clinic Institutional Review Board.

We randomly sampled 10,000 notes and then derived the unique sentences from these notes. This yielded a test set of 172,102 sentences, which were then annotated by six Mayo Clinic nurse abstractors to create a ground truth label for every word and/or phrase. Each sentence was annotated by at least two different nurse abstractors and the inter-annotator agreement on labelling a token as PHI had a Cohen’s Kappa of 0.9694 (see Supplementary Methods for details).

We also selected an additional set of 10,000 notes to fine-tune the models. Specifically, we manually annotated 61,800 unique sentences from these notes to create a tagged fine-tuning set. See Supplementary Methods for more details.

Results

We first compare the performance of the nference de-identification system with other methods on the I2B2 2014 dataset. The resulting models are evaluated using precision, recall and F1-scores (formulation provided in Supplementary Methods) for NER on several groups of PHI as defined in Table **1**. We then compare the performance of these models on the Mayo validation dataset and perform a deeper dive into the types of errors, distribution of errors per physician note and the distribution of errors per note type. Finally, we highlight the generalizability of our system on clinical document types beyond physician notes.

Group Name	Included Entities
A (defined by the HIPAA Safe Harbor Implementation)	Age over 89, Phone/Fax numbers, Email addresses, Websites and URLs, IP Addresses, Dates, Social security numbers, Medical record numbers, Vehicle/Device numbers, Account/Certificate/License numbers, Health plan numbers, Biometric identifiers, Street addresses, City, Zip code, Employer names, Personal names of patients and family members
B	Group A, Doctor names, User IDs (of care providers), State
C	Group B, Hospital names, Country
D	Group C, Holidays, Days of the week, Occupations

***Table 1**: The list of entities covered by each group of direct and quasi-identifiers. It should be noted that groups B, C and D encompass entities beyond HIPAA Safe Harbor.*

Performance on the 2014 I2B2 de-identification dataset

We compared the performance of our approach on the 2014 I2B2 test dataset with six other established de-identification tools: Scrubber [40], Physionet [41], Philter [22], MIST [42] and a model proposed by Dernoncourt et al. [27] that blends CRFs and artificial neural networks (ANNs) .

The results are provided in **Table 2**. Here, we directly report the results for Scrubber, Physionet, and Philter from existing publications [13] since the dataset (2014 I2B2) and the set of PHI entities is identical in our comparison. We trained MIST using sentences from the I2B2 training corpus (see Supplementary Methods and **Supplementary Table 3**). We present the performance of these methods on group B (see **Table 1**) entities which were their best and comparable. Finally, we cite the CRF+ANN approach [27] scores against the group D entities as we were not able to find the source code to evaluate it ourselves on group B entities.

We present two versions of the nference system. The first version was fine-tuned only on Mayo data and did not utilize any characteristics of the I2B2 training data. When evaluated with group B, this model achieved a precision, recall, and F1 score of 0.961, 0.988, and 0.974, respectively. The second version of our system involved fine-tuning our model with sentences from the I2B2 training set. We could not incorporate inclusion lists and sentence templates associated with the I2B2 data since the dataset is small. The precision, recall, and F1 increased to 0.979, 0.992, and 0.985, respectively. Precision and recall per identifier type is provided in **Supplementary Table 4**.

Method Name	Group	Precision	Recall	F1	Basis of Results
Scrubber	B	0.762	0.878	0.815	[22]
Physionet	B	0.894	0.698	0.784	[22]
Philter	B	0.785	**0.999**	0.879	[22]
MIST (Trained on I2B2)	B	0.907	0.879	0.893	N/A
nference (Fine-tuned on Mayo)	B	0.961	0.988	0.974	N/A
nference (Fine-tuned on Mayo+I2B2)	B	**0.979**	0.992	**0.985**	N/A
CRF + ANN (Dernoncourt et al.)	D	**0.979**	**0.978**	**0.978**	[27]

***Table 2**: Performance of de-identification methods on the 2014 I2B2 test corpus. The results for Scrubber, Physionet and Philter are cited from existing work since the dataset and entities for NER are the same. The results for CRF+ANN are cited from an existing publication since the source code was not available. The MIST method required training and, thus, was trained on the 2014 I2B2 training dataset. The two versions of the nference approach were fine-tuned on (i) only the Mayo dataset and (ii) both the Mayo and I2B2 datasets.*

Performance on the Mayo validation dataset

The evaluation performed on the Mayo validation dataset was based on identifiers defined by group C since this group best represented the distribution of PHI in the dataset. The performance of the de-identification methods (in terms of precision, recall and F1) are presented in **Table 3**. The nference method performed best with precision, recall, and F1 scores of 0.967, 0.994, and 0.979, respectively. Compared to the performance on the I2B2 dataset, we see improved recall (increase of 0.01) and a reduced precision value (decrease of 0.021). The F1 scores of Scrubber, Physionet and Philter were lower than those achieved on the I2B2 dataset. Among these three methods, Philter demonstrates a relatively high recall of 0.918. Closely following Philter, the MIST model achieves a recall of 0.889 with overall performance similar to that on the I2B2 dataset.

Method	Precision	Recall	F1
Scrubber	0.756	0.677	0.715
Physionet	0.837	0.772	0.803
Philter	0.709	0.918	0.800
MIST (Trained on Mayo)	0.818	0.889	0.852
nference (Fine-tuned on Mayo)	**0.967**	**0.994**	**0.979**

***Table 3**: Precision, Recall and F1-Score of various de-identification methods on the Mayo validation dataset. These methods were evaluated against group C entities.*

Error analysis on the Mayo dataset

We further investigated cases in the Mayo dataset where the nference de-identification model failed to successfully detect the PHI element completely (i.e., false negatives). This occurred at a rate of 0.6% (see **Table 4**). Across the 10,000 notes considered in the validation set, there were 848 error instances that contained these false negative errors. Accounting for duplicate occurrences of the same sentence, there were 797 unique error instances. We grouped these instances based on the type of identifier. The prevalence of the error category is shown in the second column while the third column in the table represents the contribution of each category to the error in recall (sums to 0.6%).

Category	Number of error instances (N=797)	Contribution to recall error (E=0.6%)	Example *(The PHI presented in these examples are fictitious)*
Clinic Location	208	0.1461%	He had a DWI in January and was required to do treatment through **Samson rehab** in St. Louis, Missouri
Doctor/nurse name/initial	169	0.1187%	Sent: 2020-10-20 10:00 AM Subject: RE: Consumer/**Pat**
Time (w/o date)	136	0.0955%	Following that PT session, she has another session with Dr. Smith a physical therapist, for her left foot problem at **10 o clock.**
Pharmacy Name	54	0.0379%	S: Fax received from **Trioki Rx** with request for new RX for Viread (tenofovir)

Phone Number	50	0.0351%	Phone number patient/caller is calling from or the number of the provider: **724.161.1754**.
Dates	47	0.0330%	CPL dated**4/27/04**.
Organization/Company	35	0.0246%	Last we talked about her involvement in a group called **GO GIRLS!**
Healthcare Organization	22	0.0154%	Jane is brought in by a **Minerva** female attendant and said Jane has been like this for "weeks and weeks."
Numeric Identifier	9	0.0063%	Manufactured by Merck lot number **78-32-DK**, expiration date 2020/10/20
Location (Address or partial address)	8	0.0056%	500 **State Highway 72**
Patient Name	4	0.0028%	PLOF: **X** was independent with self cares living

Table 4*: Prevalence and examples of types of false negatives encountered by the nference de-identification system when applied on the Mayo validation set. The entity highlighted in bold indicates the word or phrase that the system failed to detect.*

The most prevalent error was in the recognition of entities pertaining to clinic locations (208 out of 797). Many of these were due to partially identified phrases (e.g., "*Room 7A*" was missed in "Out of *Southwest Building Room 7A"*). The second most prevalent error was in doctor/nurse names and initials was the second highest in prevalence with 169 false negatives. Abbreviations and shorthand used by providers (typically while signing off on a clinical note) contributed to the errors in this category.

Ambiguous instances of PHI also resulted in false negatives. These were cases that a human reader would have difficulty/uncertainty in deeming as PHI. An example of this is the word *tp* in the phrase "*Comment: 03-12-2005 08:04:12 - verified tp*". We found that 26% of errors were those in which the nurse abstractors themselves did not agree on the characterization of PHI(Cohen's Kappa for errors was lower than non-errors, at 0.7453), pointing to the inherent ambiguity.

Distribution of errors per note

We further investigated the rate at which false negatives occurred on a per document level. This is because re-identification risk is directly correlated with the number of false negatives. As

shown in **Table 5**, the error instances were distributed across 637 notes. Furthermore, we see that a majority of false negatives are spread evenly across the documents. For each subsequent error rate, we computed the updated recall excluding all errors below that rate. For instance, a large majority of notes contain a single error (525 out of 637, or 82.4%) in isolation. If we exclude these singleton errors, we obtain an updated recall rate of 0.9978.

Errors per Note	Number of Notes	Total Errors	Cumulative Errors	Total Coverage	Average Number of Error Types
1	525	525	525	0.9978	1.00
2	80	160	685	0.9989	1.56
3	10	30	715	0.9991	1.75
4	6	24	739	0.9992	2.3
5	6	30	769	0.9994	2.75
6	2	12	781	0.9995	2.5
7	2	14	795	0.9996	2.25
8	2	16	811	0.9997	2.25
9	3	27	838	0.9999	2.33
10	1	10	848	1	2.0

Table 5*: Distribution of number of errors per note. Total coverage represents the updated recall rate which is computed by excluding errors below the corresponding error rate. Average number of error types denotes the number of distinct errors types (such as date errors, name errors etc) per note.*

Even for notes with a large number of errors (more than six), the number of distinct error types is between two and three. This illustrates that most of the errors are of the same type and an artifact of repetition of text within a note. For example, in the note with ten errors, eight of the instances were related to location while the remaining two are related to date. Examples of the errors pertaining to location here are "*Location of INR sample : Other: Smallville Other: Smallville Other: Smallville*", "*Recommend Recheck : Other: B-DATE Smallville Other: B-DATE Smallville", "Recommend Recheck : Other: B-DATE Other: B-DATE Smallville Other: B-DATE Other: B-DATE Smallville".* The set of location errors all pertain to the same location "Smallville" reducing the effective amount of identifiable content.

Distribution of note types

In the Mayo validation set, a physician note is associated with a note type (e.g. progress note, emergency visit, telephone encounter). Given that the structure and semantics of these note

types vary greatly from each other we analyze the enrichment of errors across them. From the 637 notes with errors, we found 134 distinct note types with at least 1 error. The top 15 note types with highest error content are listed in **Table 6**. Notes of the type “Anti Coag Service Visit Summary” contain the most errors (22 out of 26 sampled notes) followed by “Electrocardiogram” (19 out of 30 sampled notes).

Note Type	Number of Error Instances	Number of PHI Instances	Number of Notes with at Least One Error	Total Number of Notes	Fraction of Notes With at Least One Error
Ambulatory Patient Summary	59	14502	49	334	0.15
Physician Office/Clinic Message	42	8352	36	661	0.05
Report	50	3173	36	131	0.27
Medication Renewal/Refill	36	4626	31	358	0.09
Phone Message_Call	31	4840	30	391	0.08
Phone Message/Call	29	2626	24	214	0.11
Progress Note, Family Practice	27	4975	24	237	0.10
Ambulatory Discharge Medication List	27	8109	23	226	0.10
Anti Coag Service Visit Summary	24	1189	22	26	0.85
Electrocardiogram	19	411	19	30	0.63
Anticoagulation Patient Intake - Text	49	5777	18	50	0.36
Letter	15	3519	14	157	0.09
Ambulatory Depart Summary	12	3938	12	163	0.07
Progress Notes	14	3943	11	199	0.06
Telephone Encounter	12	2034	11	273	0.04

***Table 6**: Distribution of number of errors per note type. The proportion of sampled notes for a given type that contain at least one error is presented in the last column. This indicates which note types errors are more likely to occur in.*

Discussion and Future Work

Though the nference de-identification tool exhibits high levels of recall and precision, there are opportunities to boost the performance to even higher levels.

First, existing knowledge graphs and domain-specific models can be leveraged. A knowledge graph provides associations to any input term ordered by the strength of their relationship. Additionally, it groups such associations into certain categories. For example, it is expected that the top diseases associated with “ECOG” in a knowledge graph built from publicly available literature would be terms such as “nsclc” (non-small cell lung cancer) or “mcl” (mantle cell lymphoma). In the de-identification process, this could be used to recover biological terms incorrectly tagged as PHI (false positives). If strong biological associations returned by the knowledge graph for a term occur in its vicinity in the patient note then the term essentially is not PHI. As an example, if a patient’s note contains the sentences “Patient diagnosed with mantle cell lymphoma” and “ECOG performance status was determined to be 2”, ECOG would not be treated as PHI - even if the other methods detected it as PHI. In addition to knowledge graphs, existing language models trained on scientific and biological literature, with corpus specific vocabularies, can be used to further improve performance. Fine-tuning of models trained on biological and medical literature will help incorporate domain-specific linguistic features into the model. This would help improve precision by reducing false positives (biological terms being tagged as PHI); In the sentence *“There was no evidence of Bankart lesion”,* we would want to capture *“Bankart”* as a medical signal. It should not be treated as PHI (misinterpreted as name).

Second, the quality of sentences that are inputted to the model can be improved. Unstructured clinical text does not always contain well-formatted text. Punctuations as well as correct casing are commonly missing. Consider the sentence, “john fell down he got up and brushed off his leg he then called Mayo”. Filling in punctuation is likely to improve sentence tokenization and thereby boost performance on any downstream task. A case-sensitive pre-trained model along with a masked-language model objective can be used to train a system capable of correctly introducing punctuation in the right location. However, transformer models can still underperform in situations where the sentence either lacks context or is very short. Such short fragments are common in clinical documents. Bullets points for diagnosis/medications and doctor/nurse initials are two examples where sentences do not have supporting context. In such situations expanding the context using preceding and succeeding sentences may help. Document level transformer models such as Transformer-XL [43] and its variants do not have the constraint of a fixed-length context and can be explored as well.,

Third, unsupervised methods can be incorporated to accelerate the annotation process. A transformer model with fixed size vocabulary contains word representations (vectors) that are linearly independent. Grouping these vectors together yields informative clusters that can be annotated according to the nature of words present in the cluster. For example, there could be a cluster for *Names* containing words such as “*John, Jack, Mary etc*” but also including words such as “*he, she, etc*”. Similarly, a cluster for diseases may contain “*cancer, diabetes, covid, etc*”. Given these groups, the NER task can be broken down into two steps. The first involves the mask language task, where the model attempts to predict a missing word. The model

generates a list of potential candidates for the missing word from the same fixed size vocabulary. The overlap of this list with the clusters will reveal which entity type is likely to be represented by the list. In this paradigm, introducing a new entity type will not require annotation and re-training of the NER model. It will simply require marking the appropriate cluster (which is in any case a one-time task).

Conclusion

This work implemented an ensemble approach to de-identification of unstructured EHR data incorporating transformer models supported by heuristics for automatically identifying PHI across diverse clinical note types. Upon detection, suitable surrogates replaced PHI in the processed text thereby concealing residual identifiers (hiding in plain sight). The system demonstrates high precision and recall on both publicly available datasets and a large and diverse dataset from the Mayo Clinic.

Acknowledgements

We would like to thank the Mayo Clinic and the Mayo Clinic IRB under whose auspices the development of the de-identification methods and testing against real world datasets was made possible. We thank the nurse abstractors - Wendy Gay, Kathy Richmond, Denise Herman, and Sandra Severson, Dawn Pereda and Jane Emerson - for annotating the ground truth for the 172,102 sentences in the Mayo dataset that was used for testing the performance of the system, the Mayo Data Team of Ahmed Hadad, Connie Nehls and Salena Tong for preparing and helping us understand the Mayo EHR data and Andy Danielsen for supporting the collaboration. Finally, we thank Murali Aravamudan, Rakesh Barve and A. J. Venkatakrishnan for their thoughtful review and feedback on the manuscript.

Disclosures

Jeff R. Anderson, John D. Halamka, and William A. Faubion Jr. do not have any conflicts of interest in this project. Bradley Malin is a contracted consultant of the Mayo Clinic. Karthik Murugadoss, Ajit Rajasekharan, Vineet Agarwal, Sairam Bade, Jason L. Ross, Venky Soundararajan, and Sankar Ardhanari are employees of and have a financial interest in nference. Mayo Clinic and nference may stand to gain financially from the successful outcome of the research.

References

[1] Wagner, Tyler, et al. "Augmented curation of clinical notes from a massive EHR system reveals symptoms of impending COVID-19 diagnosis." *Elife* 9 (2020): e58227.

[2] Iqbal, Ehtesham, et al. "ADEPt, a semantically-enriched pipeline for extracting adverse drug events from free-text electronic health records." *PloS one* 12.11 (2017): e0187121.

[3] Jung, Kenneth, et al. “Automated detection of off-label drug use.: PLoS ONE 9, e89324 (2014).

[4] Afzal, Naveed et al. “Surveillance of peripheral arterial disease cases using natural language processing of clinical notes.” AMIA Jt Summits Transl. Sci. Proc. 2017, 28–36 (2017).

[5] Finlayson, Samuel. G. et al. “Building the graph of medicine from millions of clinical narratives.” Sci. Data 1, 140032 (2014).

[6] Garfinkel, Simson. L. "De-identification of personal information." *National institute of standards and technology* (2015).

[7] Berg, Hanna et al. "The Impact of De-identification on Downstream Named Entity Recognition in Clinical Text." *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*. 2020.

[8] Leevy, Joffrey L., Taghi M. Khoshgoftaar, and Flavio Villanustre. "Survey on RNN and CRF models for de-identification of medical free text." *Journal of Big Data* 7.1 (2020): 1-22.

[9] Yogarajan, Vithya, et al."A survey of automatic de-identification of longitudinal clinical narratives." *arXiv preprint arXiv:1810.06765* (2018).

[10] Yogarajan, Vithya, et al. "Automatic end-to-end De-identification: Is high accuracy the only metric?." *arXiv preprint arXiv:1901.10583* (2019).

[11] Neamatullah, Ishna, et al. "Automated de-identification of free-text medical records." *BMC medical informatics and decision making* 8.1 (2008): 32.

[12] Douglass, Margaret, et al. "De-identification algorithm for free-text nursing notes." *Computers in Cardiology, 2005*. IEEE, 2005.

[13] Douglass, Margaret, et al. "Computer-assisted de-identification of free text in the MIMIC II database." *Computers in Cardiology, 2004*. IEEE, 2004.

[14] Yang, Zhilin, et al. "XLNet: Generalized Autoregressive Pre-training for Language Understanding." *arXiv preprint arXiv:1906.08237* (2019).

[15] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017.

[16] Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Neural machine translation of rare words with subword units." *arXiv preprint arXiv:1508.07909* (2015).

[17] Wu, Yonghui, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." *arXiv preprint arXiv:1609.08144* (2016).

[18] Kudo, Taku, and John Richardson. "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing." *arXiv preprint arXiv:1808.06226* (2018).

[19] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).

[20] Morrison, Frances P. et al. "Repurposing the clinical record: can an existing natural language processing system de-identify clinical notes?." *Journal of the American Medical Informatics Association* 16.1 (2009): 37-39.

[21] Uzuner, Özlem et al. "Evaluating the state-of-the-art in automatic de-identification." *Journal of the American Medical Informatics Association* 14.5 (2007): 550-563.

[22] Norgeot, Beau, et al. "Protected Health Information filter (Philter): accurately and securely de-identifying free-text clinical notes." *NPJ digital medicine* 3.1 (2020): 1-8.

[23] Ruch, Patrick, et al. "Medical document anonymization with a semantic lexicon." *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2000.

[24] Ferrández, Oscar, et al. "Evaluating current automatic de-identification methods with Veteran's health administration clinical documents." *BMC medical research methodology* 12.1 (2012): 109.

[25] Meystre, Stephane M., et al. "Automatic de-identification of textual documents in the electronic health record: a review of recent research." *BMC medical research methodology* 10.1 (2010): 70.

[26] Li, Muqun, et al. "Efficient active learning for electronic medical record de-identification." *AMIA Summits on Translational Science Proceedings* 2019 (2019): 462.

[27] Dernoncourt, Frank, et al. "De-identification of patient notes with recurrent neural networks," CoRR, 2016.

[28] Khin, Kaung, Philipp Burckhardt, and Rema Padman. "A deep learning architecture for de-identification of patient notes: implementation and evaluation." *arXiv preprint arXiv:1810.01570* (2018).

[29] Popel, Martin, et al. "Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals." *Nature communications* 11.1 (2020): 1-15.

[30] Stubbs, Amber, and Özlem Uzuner. "Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/UTHealth corpus." *Journal of biomedical informatics* 58 (2015): S20-S29.

[31] Z. Liu, et al. “Automatic de-identification of electronic medical records using token-level and character-level conditional random fields,” Journal of Biomedical Informatics, vol. 58, no. nil, pp. S47–S52, 2015.

[32] Kim, Youngjun et al. "Ensemble-based methods to improve de-identification of electronic health record narratives." *AMIA Annual Symposium Proceedings*. Vol. 2018. American Medical Informatics Association, 2018.

[33] Kim, Youngjun, and Stéphane M. Meystre. "Ensemble method–based extraction of medication and related information from clinical texts." *Journal of the American Medical Informatics Association* 27.1 (2020): 31-38.

[34] Kim, Youngjun, and Stéphane M. Meystre. "Voting Ensemble Pruning for De-identification of Electronic Health Records."

[35] Lee, Hee-Jin, et al. "A hybrid approach to automatic de-identification of psychiatric notes." *Journal of biomedical informatics* 75 (2017): S19-S27.

[36] Ferrández, Oscar, et al. "A hybrid stepwise approach for de-identifying person names in clinical documents." *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*. 2012.

[37] Sweeney, Latanya. "Replacing personally-identifying information in medical records, the Scrub system." *Proceedings of the AMIA annual fall symposium*. American Medical Informatics Association, 1996.

[38] Carrell, David, et al. "Hiding in plain sight: use of realistic surrogates to reduce exposure of protected health information in clinical text." *Journal of the American Medical Informatics Association* 20.2 (2013: 342-348.

[39] Carrell, David S., et al. "The machine giveth and the machine taketh away: a parrot attack on clinical text deidentified with hiding in plain sight." *Journal of the American Medical Informatics Association* 26.12 (2019): 1536-1544.

[40] McMurry, Andrew J., et al. "Improved de-identification of physician notes through integrative modeling of both public and private medical text." *BMC Medical Informatics and Decision making* 13.1 (2013): 112.

[41] Neamatullah, Ishna, et al. "Automated de-identification of free-text medical records." *BMC Medical Informatics and Decision making* 8.1 (2008): 32.

[42] Aberdeen, John, et al. "The MITRE Identification Scrubber Toolkit: design, training, and assessment." *International Journal of Medical Informatics* 79.12 (2010): 849-859.

[43] Dai, Zihang, et al. "Transformer-XL: Attentive language models beyond a fixed-length context." *arXiv preprint arXiv:1901.02860* (2019).

Supplementary Methods

Model fine-tuning details

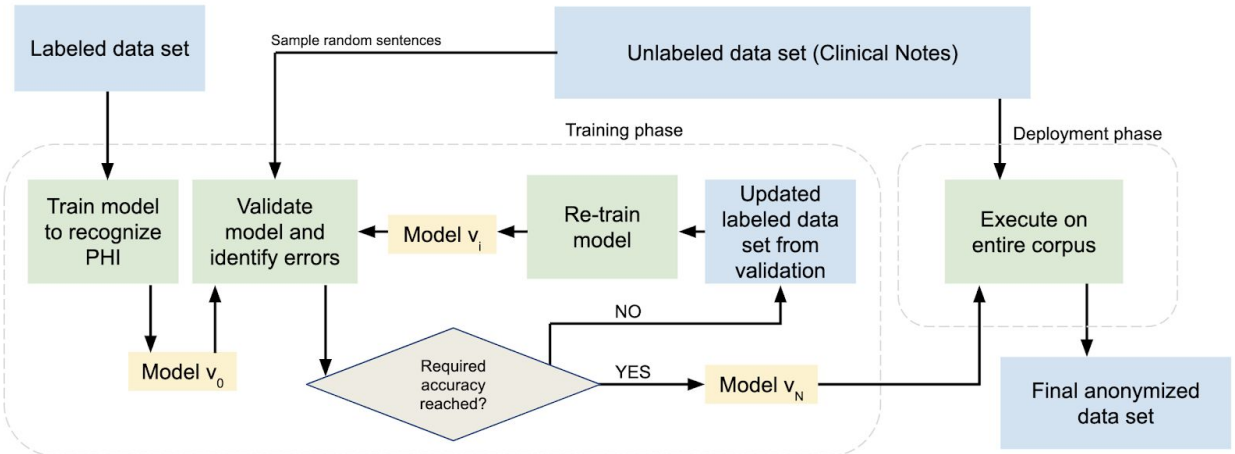
We employed the *bert-base-cased* model (https://huggingface.co/bert-base-cased) through the HuggingFace/Transformers (https://github.com/huggingface/transformers) library. This is a case-sensitive English language pre-trained model based off of the BERT architecture trained using a masked language modelling (MLM) objective. The BERT model was pretrained on BookCorpus (https://huggingface.co/datasets/bookcorpus), a dataset comprising 11,038 unpublished books in addition to English Wikipedia. Our ensemble involved employing an individual model per PHI entity type. Each model was fine-tuned using a total of 61,800 tagged sentences. The final number of examples for each entity type is shown in the table below.

Entity Type	Fine-tuning Examples
Name	44,929
Organization	44,825
Location	11,461
Age	5,409

Supplementary Table 1: *Number of fine-tuning examples for each entity type*

Each transformer model is iteratively fine-tuned with training samples being continuously added to the initial set of training samples. The sentences chosen for fine-tuning the model are specifically selected from the space of errors that was seen in prior models. The iterative process of fine-tuning models therefore results in the generation of multiple individual neural networks (different versions) for each PHI type each having a specific performance. To maximize the overall recall, we choose the two best performing models for each entity type and employ them in tandem. More specifically, when an entity is detected by either of these two models, it is considered to be PHI.

To complement the above improvements on model architecture and algorithms for de-identification, an iterative learning framework is deployed in tandem that allows rapid validation and performance evaluation for trained models (***Supplementary Fig. 1****)*. This allows each component of the ensemble framework to be re-trained and fine-tuned to learn from previous mistakes independently of other models.



Supplementary Fig. 1: *Iterative model generation process and learning from errors. Model performance improves during its evolution from v0 to vN.*

Creating an inclusion list of sentences

In a repository of 103 million physician notes (from 477,000 patients) from the Mayo Clinic, a total of approximately 3.1 billion sentences corresponded to approximately 700 million unique sentences, which highlights the redundancy in a corpus of this size and provides optimization opportunities in the de-identification processing pipeline. In particular, sentences with high prevalence were found to typically not contain PHI (since they occur across a large number of patients, the chances that they contain information specific to any one patient is low). We computed the prevalence of all sentences and found that the top 1,600 most common sentences correspond to 1.01 billion sentences overall (one-third of the entire corpus).

These 1,600 sentences represented the initial inclusion list. Additionally, we filtered out the top 25,000 most prevalent sentences that contain a disease or a drug entity. This ensures that medically relevant sentences that are also highly prevalent are preserved. All of the sentences that are part of the inclusion list are manually verified.

Obfuscation methods

For each category of PHI, obfuscation is performed through the replacement methods described in **Supplementary Table 2**.

Category	Sub-category	Replacement Method	Example
Name	First Name	Replace with sampled surrogate after gender and ethnicity matching	**Mohammad** visited the clinical today. → **Imran** visited the clinic today.
Name	Last Name	Replace with sampled surrogate after ethnicity matching	Ms. **Lopez** agreed with the procedure → Ms. **Hernandez** agreed with the

			procedure.
Name	Initial	Replace letters randomly	John **W.B.** Smith → Jack **G.S.** Parker
Name	IDs	Replace letters and numbers randomly	Signed **DF14** → Signed **AB76**
Location	N/A	Replace with sampled surrogate	She is from **Springfield, Illinois** → She is from **Ithaca, New Yor**k
Organization	N/A	Replace with sampled surrogate	Welcome to **Veterans Memorial Center** → Welcome to **Butler County Health Care Center**
Age	N/A	If age is greater than 89 years, replace with “89+”	Mr. Johnson is **92** years old → Mr. Michaels is **89+** years old
Date	N/A	Shift date by a randomly selected number of days. Maintain format of the date string.	Appt date: **04/12/2020** → Appt date: **03/29/2020**
Time	N/A	Do nothing	N/A
Website	N/A	Replace with sampled surrogate	For more info check **mayoclinic.org** → For more info check **healthcareforyou.org**
Email Address	N/A	Replace with sampled surrogate	Reach out to **john.smith@care.com** → Reach out to **primaryprovider@care.com**
Vehicle Plate	N/A	Replace letters and numbers randomly	Vehicle plate: **6TR-435** → Vehicle plate: **7TH-129**
Phone Number	N/A	Replace numbers randomly	**546-123-0543** → **574-784-1122**
Numeric Identifier	N/A	Replace numbers randomly	Patient Clinic **#4433245** → Patient Clinic **#1382135**
Zip Code	N/A	Replace numbers randomly	Cambridge MA, **02139** → Tucson, AZ, **45241**

Pager	N/A	Replace numbers randomly	Dr. Jones **1-12435** → Dr. Smith **4-63259**
IP Address	N/A	Replace numbers randomly	**127.0.0.1** → **176.3.5.7**

***Supplementary Table 2**: Obfuscation methods for each PHI category*

Evaluation metrics

To evaluate model performance on the de-identification task, we computed the precision, recall and F1 scores. These were computed as follows:

$$\text{Precision} = TP / (TP+FP)$$

$$\text{Recall} = TP / (TP+FN)$$

$$F1 = 2*\text{Precision}*\text{Recall} / (\text{Precision}+\text{Recall})$$

where TP is the true positive count, FP is the false positive count and FN is the false negative count.

De-identification on 2014 I2B2 test dataset

Evaluation of existing methods: We evaluated Scrubber, Physionet and Philter systems on the 2014 I2B2 data in their standard modes of operation (without additional dictionaries or gazetteers). To run MIST on the 2014 I2B2 data, we converted the dataset into the 2006 I2B2 data format since the stable software release of MIST directly supported the 2006 format (and not the 2014 format). Additionally, MIST assigns PHI categories that are different from the 2014 I2B2 entity set. To address this issue, we constructed a mapping between the two sets of PHI categories as described in **Supplementary Table 3**. In our implementation of MIST, we did not use gazetteers. As a result the scores we report for MIST are lower than those of the Dernoncourt et al. implementation which was configured to use the same gazetteers as their CRF model.

MIST PHI Category	**I2B2 PHI Categories**
NAME	PATIENT, DOCTOR, USERNAME
LOCATION	ORGANIZATION, STREET, CITY, STATE, COUNTRY, ZIP, LOCATION-OTHER
AGE	AGE
DATE	DATE
CONTACT	PHONE, FAX, EMAIL

ID	IDNUM, MEDICALRECORD, DEVICE
PROFESSION	PROFESSION

Supplementary Table 3: *Mapping between MIST and I2B2 PHI categories*

Handling document IDs: The nference system was designed to identify document IDs in unstructured text (e.g. "*3-1272852*" in the sentence "*eScription document: 3-1272852 BFFocus*"). These entities were however not marked as PHI in the ground truth of the I2B2 dataset and hence contributed to the false positive rate of our system. If we exclude such cases (we found 87 instances of document ID) our precision improves from 0.979 to 0.986.

PHI entity-wise precision and recall comparison: For each entity class and I2B2 entity type we computed the precision and recall for both versions of the nference system (fine-tuned only on Mayo data and fine-tuned on Mayo as well as I2B2 data) as shown in **Supplementary Table 4**. Since the tagset used by nference groups is different from I2B2 entities, it was not possible to compute the F-Score on a per entity basis. That is, while the recall could be calculated for each I2B2 entity, the precision could only be determined at the level of the entity class. Rule-based components on the nference ensemble performed identically across both versions of our system since they are not impacted by fine-tuning.

		nference (fine-tuned on Mayo)		**nference (fine-tuned on Mayo+I2B2)**	
Entity Class	**I2B2 Entity**	**Precision**	**Recall**	**Precision**	**Recall**
Date		0.975		0.993	
	DATE		0.994		0.994
Names		0.974		0.996	
	PATIENT		0.992		0.998
	DOCTOR		0.992		0.993
	USERNAME		0.954		0.954
Location		0.911		0.968	
	STREET		0.978		0.992
	CITY		0.982		1.0
	STATE		1.0		1.0
	COUNTRY		1.0		1.0

	ZIP		1.0		1.0
	LOCATION-OTHER		0.55		0.6
Organization		0.969		0.991	
	HOSPITAL		0.821		0.922
	ORGANIZATION		0.759		0.912
Numeric Identifiers		0.926		0.926	
	IDNUM		0.968		0.968
	DEVICE		0.9		0.9
	MEDICALRECORD		0.986		0.986
Contact		1.0		1.0	
	PHONE		0.994		0.994
	FAX		0.666		0.666
	EMAIL		1.0		1.0

Supplementary Table 4: *PHI entity-wise precision and recall for both versions of the nference system: (a) Fine-tuned on Mayo and (b) Fine-tuned on Mayo+I2B2. The first column corresponds to the entity class and the second column corresponds to the specific I2B2 entity type. Dates, numeric identifiers and contacts are implemented through rule-based methods and therefore have the same precision and recall across both system versions. For this analysis, only ages over 89 in the test dataset were considered (totally 8 instances of such an age were found) and our method detected all of those entities successfully. We therefore omit ages from this table.*

Mayo validation set annotation

Inter-rater reliability

Cohen's Kappa is used to compute the inter-rater reliability for categorical terms. We calculate Cohen's Kappa for the Mayo validation dataset annotated by Mayo Clinic nurses in the following manner.

Step 1: In the ground truth tagged sentences for each nurse, we convert each PHI entity (e.g., names, dates, and locations) to a universal " PHI entity" type. Non PHI entities are left as is.
Step 2: Since the full set of sentences to review is split into three groups and within each group every sentence is reviewed by two nurses, we consider two nurse extractor groups. Group 1 is comprised of nurses 1, 3, and 5 and group 2 is comprised of nurses #2, #4, and #6.

Step 3: We then construct an agreement/disagreement matrix. The numbers in the **Supplementary Table 5** denote the number of words for each category. For example, there are 4,919 words that were marked as PHI by group 1 but were not marked as PHI by group 2.

		Nurse Extractors Group 2	
		PHI entity	Non PHI entity
Nurse Extractors Group 1	PHI entity	185455 (a)	4919 (b)
	Non PHI entity	5411 (c)	1483221 (d)

Supplementary Table 5: *Agreement matrix for measuring inter-rater reliability*

Step 4: The observed proportionate agreement p_0 = (a+d)/(a+b+c+d) = **0.9938**

Step 5: The expected probability (i.e. probability of random agreement between the two groups) is the probability that both groups agreed on either yes or no. The probability that both groups agreed on yes (p_{yes}) is given below

P_{yes} = (a+b)/(a+b+c+d) . (a+c)/(a+b+c+d) = **0.0128**

P_{no} = (c+d)/(a+b+c+d) . (b+d)/(a+b+c+d) = **0.7858**

Therefore,

$p_e = p_{yes} + p_{no}$ = **0.7987**

Step 6: Compute Cohen's Kappa

$\kappa = (p_o - p_e)/(1 - p_e)$ = **0.9694**