Journal of the American Medical Informatics Association, 24(3), 2017, 596–606
doi: 10.1093/jamia/ocw156
Advance Access Publication Date: 31 December 2016
Research and Applications

Research and Applications

De-identification of patient notes with recurrent neural networks

Franck Dernoncourt,[1,*] Ji Young Lee,[1,*] Ozlem Uzuner,[2] and Peter Szolovits[1]

[1]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA, [2]Computer Science Department, University at Albany, SUNY, Albany, NY, USA

Corresponding Author: Franck Dernoncourt, 32 Vassar St, 32-293, Cambridge, MA 02139, USA. E-mail: francky@mit.edu; Tel: +1-443-637-2659
*These authors contributed equally to this work.

Received 25 June 2016; Revised 6 September 2016; Accepted 6 October 2016

Abstract

Objective: Patient notes in electronic health records (EHRs) may contain critical information for medical investigations. However, the vast majority of medical investigators can only access de-identified notes, in order to protect the confidentiality of patients. In the United States, the Health Insurance Portability and Accountability Act (HIPAA) defines 18 types of protected health information that needs to be removed to de-identify patient notes. Manual de-identification is impractical given the size of electronic health record databases, the limited number of researchers with access to non-de-identified notes, and the frequent mistakes of human annotators. A reliable automated de-identification system would consequently be of high value.
Materials and Methods: We introduce the first de-identification system based on artificial neural networks (ANNs), which requires no handcrafted features or rules, unlike existing systems. We compare the performance of the system with state-of-the-art systems on two datasets: the i2b2 2014 de-identification challenge dataset, which is the largest publicly available de-identification dataset, and the MIMIC de-identification dataset, which we assembled and is twice as large as the i2b2 2014 dataset.
Results: Our ANN model outperforms the state-of-the-art systems. It yields an F1-score of 97.85 on the i2b2 2014 dataset, with a recall of 97.38 and a precision of 98.32, and an F1-score of 99.23 on the MIMIC de-identification dataset, with a recall of 99.25 and a precision of 99.21.
Conclusion: Our findings support the use of ANNs for de-identification of patient notes, as they show better performance than previously published systems while requiring no manual feature engineering.

Key words: medical language processing, de-identification, neural networks

INTRODUCTION AND RELATED WORK

In many countries including the United States, medical professionals are strongly encouraged to adopt electronic health records (EHRs) and may face financial penalties if they fail to do so.[1,2] The Centers for Medicare and Medicaid Services have paid out more than $30 billion in EHR incentive payments to hospitals and providers who have attested to meaningful use as of March 2015. Medical investigations can greatly benefit from the resulting increasingly large EHR datasets. One of the key components of EHRs is patient notes; the information they contain can be critical for a medical investigation, because much information present in texts cannot be found in the other elements of the EHR. However, before patient notes can be shared with medical investigators, some types of information, referred to as protected health information (PHI), must be removed in order to preserve patient confidentiality. In the United States, the Health Insurance Portability and Accountability Act (HIPAA)[3] defines 18 different

Table 1. PHI types as defined by HIPAA, i2b2, and MIMIC

PHI categories	PHI types	Descriptions	HIPAA	i2b2	MIMIC
AGE	AGE	Ages ≥90	x	x	x
		Ages <90		x	
CONTACT	PHONE	Telephone numbers	x	x	x
	FAX	Fax numbers	x	x	PHONE
	EMAIL	Electronic mail addresses	x	x	
	URL	Uniform resource locators	x	–	
	IP ADDRESS	Internet protocol addresses	x	–	
DATE	DATE	Dates (month and day parts)	x	x	x
		Year		x	x
		Holidays		x	x
		Days of the week		x	
ID	IDNUM	Social Security numbers	x	x	x
		Account numbers	x	x	x
		Certificate or license numbers	x	x	x
	MEDICAL RECORD	Medical record numbers	x	x	IDNUM
	DEVICE	Vehicle or device identifiers	x	x	IDNUM
	HEALTH PLAN	Health plan numbers	x	–	IDNUM
	BIOID	Biometric identifiers or full-face photographs	x	–	
LOCATION	STREET	Street address	x	x	x
	CITY	City	x	x	LOCATION-OTHER
	ZIP	Zip code	x	x	x
	STATE	State		x	x
	COUNTRY	Country		x	x
	LOCATION-OTHER	Other identifiable locations such as landmarks		x	x
	ORGANIZATION	Employers	x	x	
	HOSPITAL	Hospital name		x	x
		Ward name			x
NAME	PATIENT	Names of patients and family members	x	x	x
	DOCTOR	Provider name		x	x
	USERNAME	User IDs of providers		x	
PROFESSION	PROFESSION	Profession		x	

Classification of PHI into categories and types are as defined in the i2b2 dataset. During training, the PHI types are used as the labels to predict. The mark "–" denotes that two or fewer instances of the corresponding PHI types are present in the whole dataset, and no instance is present in the test set. In the MIMIC dataset, some PHI types are mapped to a different PHI type due to data ambiguity or sparsity issues: these PHI types are marked with the specific PHI type that it is mapped to instead of the mark "x".

types of PHI, ranging from patient names to phone numbers. Table 1 presents the exhaustive list of PHI types as defined by HIPAA.

The task of removing PHI from a patient note is referred to as de-identification, since the patient cannot be identified once PHI is removed. De-identification can be either manual or automated. Manual de-identification means that the PHI is labeled by human annotators. There are three main shortcomings of this approach. First, only a restricted set of individuals is allowed to access the identified patient notes, thus the task cannot be crowdsourced. Second, humans are prone to mistakes. Neamatullah et al.[4] asked 14 clinicians to detect PHI in approximately 130 patient notes; the results of the manual de-identification varied from clinician to clinician, with recall ranging from 0.63 to 0.94. Third, human annotation is costly. Douglass et al.[5,6] reported that annotators were paid US$50 per hour and read 20 000 words per hour at best.

As a matter of comparison, the MIMIC dataset,[7,8] which contains data from 50 000 intensive care unit stays, consists of 100 million words. This would require 5000 hours of annotation, which would cost US$250 000 at the same pay rate. Given the annotators' spotty performance, each patient note would have to be annotated by at least two different annotators; it would therefore cost at least US$500 000 to de-identify the notes in the MIMIC dataset.

In order to reduce the cost of annotating, many studies investigate the use of machine pre-annotation, where human annotators are provided with machine-annotated data to reduce the annotation time. Lingret et al.[9] show that using pre-annotation resulted in 13.85–21.5% in time savings for developing a clinical named-entity recognition corpus. However, another study by South et al.[10] showed that using machine pre-annotation along with an interactive annotation tool neither improved the quality nor decreased the time investment when annotating a clinical text de-identification corpus.

Instead of annotating all documents at the same time from either raw or pre-annotated texts, Hanauer et al.[11] took a novel approach, where annotations were performed alternately by humans and machine. More specifically, the clinical notes were divided into multiple batches of 10, 20, or 40 notes and each batch was annotated sequentially by human annotators after being pre-annotated by a de-identifier trained on previously annotated batches. They showed that the annotation time for each instance decreased in later batches as the de-identifier's performance improved, achieving an F1-score

Table 2. Examples of features used in the CRF model

Feature types	Features
Lexical/syntactic	Token, lemma, tense, parts of speech
Morphological	Ends with s, contains a digit, is numeric, is alphabetic, is alphanumeric, is title case, is all lowercase, prefix, suffix
Temporal	Season, month, weekday, time of day
Semantic/wordnet	Hypernyms, senses, lemma names
Gazetteers	First names, last names, medical titles, medical specialties, cities, states (including abbreviations), countries, organizations, professions, holidays
Regular expressions	E-mail, age, date, phone, zip code, ID number, medical record number

of 0.95 after just over 8 hours of annotation time (after 20 batches of 10 notes each). Similarly, Gobbel et al.[12] presented a tool called RapTAT to assist human annotators by pre-annotating the documents interactively while the annotators are working on them, resulting in up to 50% reduction in annotation time.

Automated de-identification systems can be classified into two categories: rule-based systems and machine learning-based systems. Rule-based systems[4,14–22] typically rely on patterns, expressed as regular expressions and gazetteers, defined and tuned by humans. They do not require any labeled data (aside from labels required for evaluating the system) and are easy to implement and interpret, which explains their large presence in the industry.[13] However, they need to be fine-tuned for each new dataset, are not robust to language changes (e.g., variations in word forms, typographical errors, or infrequently used abbreviations), and cannot easily take into account the context (e.g., "Mr. Parkinson" is PHI, while "Parkinson's disease" is not). To alleviate some downsides of the rule-based systems, there have been many attempts to use supervised machine learning algorithms to de-identify patient notes. These algorithms are used to train a classifier to label each word as PHI or not PHI, sometimes distinguishing between different PHI types. Common statistical methods include decision trees,[23] log-linear models, support vector machines,[24–26] and conditional random fields (CRFs).[27] The latter is employed in most state-of-the-art systems. For a thorough review of existing systems, see Meystre et al.[28] and Stubbs et al.[29] All these methods share two downsides: they require a decent-sized labeled dataset and much feature engineering. As with rules, quality features are challenging and time-consuming to develop.

Recent approaches to natural language processing based on artificial neural networks (ANNs) do not require handcrafted rules or features. Instead, ANNs can automatically learn effective features by performing composition over tokens, which are represented as vectors, often called token embeddings. The token embeddings are jointly learned with the other parameters of the ANN. They can be initialized randomly, or can be pre-trained using large unlabeled datasets typically based on token co-occurrences.[30–32] The latter often performs better, since the pre-trained token embeddings explicitly encode many linguistic regularities and patterns. As a result, methods based on ANNs have shown promising results for various tasks in natural language processing (NLP), such as language modeling,[33] text classification,[34–37] question answering,[38,39] machine translation,[40–42] and named entity recognition.[31,43,44] A few methods also use vector representations of characters as inputs in order to either replace or augment token embeddings.[43–45]

Inspired by the performance of ANNs for various other NLP tasks, this article introduces the first de-identification system based on ANNs. Unlike other machine learning-based systems, ANNs do not require manually curated features, such as those based on regular expressions and gazetteers. We show that ANNs achieve state-of-the-art results on de-identification of two different datasets for patient notes, the i2b2 2014 challenge dataset and the MIMIC dataset. To the best of our knowledge, this is the first paper to introduce ANN-based approaches using token and character embeddings to the clinical de-identification task.

A few related publications have applied ANNs and word embeddings for clinical NLP tasks. Wu et al.[46] investigated the use of deep neural networks to learn word embeddings and perform named entity recognition of four types of clinical entities – problems, lab tests, procedures, and medications – on Chinese clinical text. Two submissions[47,48] to a recent SemEval-2016 Task 12: Clinical TempEval challenge report ANN-based methods for information extraction from clinical notes and pathology reports. Li and Huang[47] used convolutional neural networks and Fries[48] compared the performance of recurrent neural networks (RNNs) and DeepDive[49] for the task.

METHODS AND MATERIALS

We first present a de-identifier we developed based on a CRF model. This de-identifier yielded state-of-the-art results with the i2b2 2014 dataset, which is the reference dataset for comparing de-identification systems. This system is used as a challenging baseline for the ANN model that we present next. The ANN model outperformed the CRF model, as outlined in the results section.

CRF model

In the CRF model, each patient note is tokenized using the Stanford CoreNLP tokenizer,[50] and features are extracted for each token. During the training phase, the CRF's parameters are optimized to maximize the likelihood of the gold standard labels. During the test phase, the CRF predicts the labels. The performance of a CRF model depends mostly on the quality of its features. We used a combination of lexical, morphological, temporal, semantic, gazetteer, and regular expression features. Table 2 lists some of the features used in the CRF model. The regular expressions were written mostly based on the best-performing CRF-based competitors in the i2b2 challenge.[51] The gazetteers were compiled using common resources from the Web, and most other features were from Filannino et al.[52]

In order to effectively incorporate context when predicting a label, all the features for a given token are computed based on that token and on the four surrounding tokens.

ANN model

The main components of the ANN model are RNNs. In particular, we use a type of RNN called long short-term memory (LSTM),[53] as discussed in the following subsection. The system is composed of 3 layers:

- Character-enhanced token-embedding layer
- Label prediction layer and
- Label-sequence optimization layer.

As in the CRF model, patient notes are first tokenized using the Stanford CoreNLP tokenizer. The character-enhanced token-embedding layer maps each token to a vector representation. The sequence of vector representations corresponding to a sequence of tokens is inputted into the label-prediction layer, which outputs the sequence of vectors containing the probability of each label for each corresponding token. Lastly, the sequence-optimization layer outputs the most likely sequence of predicted labels based on the sequence of probability vectors from the previous layer. All layers are learned jointly. Figure 1 shows the ANN architecture.

In the following, we denote scalars in italic lowercase (e.g., k, b_f), vectors in bold lowercase (e.g., $\mathbf{s}$, $\mathbf{x}_i$), and matrices in italic uppercase (e.g., W_f) symbols. We use the colon notations $x_{i:j}$ and $\mathbf{v}_{i:j}$ to denote the sequence of scalars $x_i, x_{i+1}, \ldots, x_j$ and vectors $\mathbf{v}_i, \mathbf{v}_{i+1}, \ldots, \mathbf{v}_j$, respectively.

Bidirectional LSTM

An RNN is a neural network architecture designed to handle input sequences of variable sizes, but it fails to model long-term dependencies. An LSTM is a type of RNN that mitigates this issue by keeping a memory cell that serves as a summary of the preceding elements of an input sequence. More specifically, given a sequence of vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, at each step $t = 1, \ldots, n$, an LSTM takes as input $\mathbf{x}_t$, $\mathbf{h}_{t-1}, \mathbf{c}_{t-1}$ and produces the hidden state $\mathbf{h}_t$ and the memory cell $\mathbf{c}_t$ based on the following formulas:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(W_i\,[\mathbf{x}_t;\ \mathbf{h}_{t-1};\ \mathbf{c}_{t-1}] + \mathbf{b}_i)\\
\tilde{\mathbf{c}}_t &= \tanh(W_c\,[\mathbf{x}_t;\ \mathbf{h}_{t-1}] + \mathbf{b}_c)\\
\mathbf{c}_t &= (1 - \mathbf{i}_t) \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t\\
\mathbf{o}_t &= \sigma(W_o\,[\mathbf{x}_t;\ \mathbf{h}_{t-1};\ \mathbf{c}_t] + \mathbf{b}_o)\\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t)
\end{aligned}$$

where W_i, W_c, W_o are weight matrices and $\mathbf{b}_i, \mathbf{b}_c, \mathbf{b}_o$ are bias vectors used in the input gate, memory cell, and output gate calculations, respectively. The symbols $\sigma(\cdot)$ and $\tanh(\cdot)$ refer to the element-wise sigmoid and hyperbolic tangent functions, and $\odot$ is the element-wise multiplication. $\mathbf{h}_0 = \mathbf{c}_0 = 0$.

A bidirectional LSTM consists of a forward LSTM, which calculates the forward hidden states $(\overrightarrow{\mathbf{h}}_1, \overrightarrow{\mathbf{h}}_2, \ldots, \overrightarrow{\mathbf{h}}_n)$, and a backward LSTM, which calculates the backward hidden states $(\overleftarrow{\mathbf{h}}_1, \overleftarrow{\mathbf{h}}_2, \ldots, \overleftarrow{\mathbf{h}}_n)$ by feeding the input sequence in backward order, from $\mathbf{x}_n$ to $\mathbf{x}_1$.

Depending on the application of the LSTM, one might need an output sequence corresponding to each element in the sequence, or a single output that summarizes the whole sequence. In the former case, the output sequence $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n$ of the LSTM is obtained by concatenating the hidden states of the forward and backward LSTMs for each element, i.e., $\overleftrightarrow{\mathbf{h}}_t = \left(\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t\right)$ for $t = 1, \ldots, n$. In the latter case, the output is obtained by concatenating the last hidden states of the forward and backward LSTMs, i.e., $\overleftrightarrow{\mathbf{h}}_t = (\overrightarrow{\mathbf{h}}_n; \overleftarrow{\mathbf{h}}_n)$.

Character-enhanced token embedding layer

The character-enhanced token-embedding layer takes a token as input and outputs its vector representation. The latter results from the

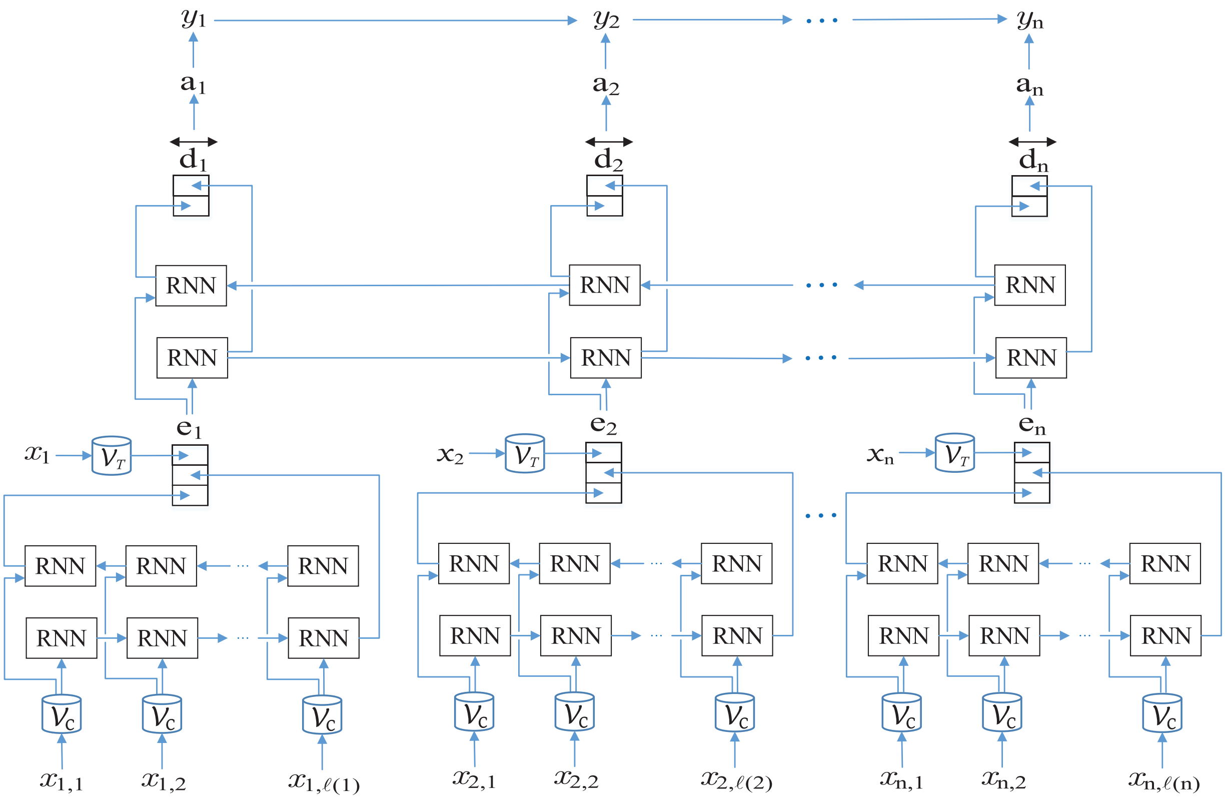

Figure 1. Architecture of the artificial neural network (ANN) model. (RNN, recurrent neural network.) The type of RNN used in this model is long short-term memory (LSTM). n is the number of tokens, and x_i is the i^{th} token. $\mathcal{V}_T$ is the mapping from tokens to token embeddings. $\ell(i)$ is the number of characters and $x_{i,j}$ is the j^{th} character in the i^{th} token. $\mathcal{V}_C$ is the mapping from characters to character embeddings. $\mathbf{e}_i$ is the character-enhanced token embeddings of the i^{th} token. $\overleftrightarrow{\mathbf{d}}_i$ is the output of the LSTM of the label prediction layer, $\mathbf{a}_i$ is the probability vector over labels, y_i is the predicted label of the i^{th} token.

concatenation of two different types of embeddings; the first one directly maps a token to a vector, while the second one comes from the output of a character-level token encoder. The direct mapping $\mathcal{V}_T(\cdot)$ from token to vector, often called a token (or word) embedding, can be pre-trained on large unlabeled datasets using programs such as word2vec[30,54,55] or GloVe,[32] and can be learned jointly with the rest of the model. Token embeddings, often learned by sampling token co-occurrence distributions, have desirable properties, such as locating semantically similar words closely in the vector space, leading to state-of-the-art performance for various tasks.

While the token embeddings capture the semantics of tokens to some degree, they may still suffer from data sparsity. For example, they cannot account for out-of-vocabulary tokens, misspellings, and different noun forms or verb endings. One solution to remediate some of these issues would be to lemmatize tokens before training, but this approach may fail to retain some useful information, such as the distinctions between some verb and noun forms.

We address this issue by using character-based token embeddings, which incorporate each individual character of a token to generate its vector representation. This approach enables the model to learn sub-token patterns such as morphemes (e.g., suffixes or prefixes) and roots, thereby capturing out-of-vocabulary tokens, different surface forms, and other information not contained in the token embeddings.

Let $x_{i,1},\ldots,x_{i,\ell(i)}$ be the sequence of characters that comprise the i^{th} token x_i, where $\ell(i)$ is the number of characters in x_i. The character-level token encoder generates the character-based token embedding of x_i by first mapping each character $x_{i,j}$ to a vector $\mathcal{V}_C(x_{i,j})$, called a character embedding, via the mapping $\mathcal{V}_C(\cdot)$. Then the sequence $\mathcal{V}_C(x_{i,1}),\ \ldots,\ \mathcal{V}_C(x_{i,\ell(i)})$ is passed to a bidirectional LSTM, which outputs the character-based token embedding $\overleftrightarrow{\mathbf{b}}_i$.

As a result, the final output $\mathbf{e}_i$ of the character-enhanced token-embedding layer for i^{th} token x_i is the concatenation of the token embedding $\mathcal{V}_T(x_i)$ and the character-based token embedding $\overleftrightarrow{\mathbf{b}}_i$ In summary, when the character-enhanced token-embedding layer receives a sequence of tokens $x_{1:n}$ as input, it will output the sequence of token embeddings $\mathbf{e}_{1:n}$.

Label-prediction layer

The label-prediction layer takes as input the sequence of vectors $\mathbf{e}_{1:n}$, i.e., the outputs of the character-enhanced token-embedding layer, and outputs $\mathbf{a}_{1:n}$, where the t^{th} element of $\mathbf{a}_n$ is the probability that the n^{th} token has the label t. The label is either one of the PHI types or non-PHI. For example, if we aimed to predict all 18 HIPAA-defined PHI types, there would be 19 different labels.

The label-prediction layer contains a bidirectional LSTM that takes the input sequence $\mathbf{e}_{1:n}$ and generates the corresponding output sequence $\overleftrightarrow{\mathbf{d}}_{i:n}$. Each output $\overleftrightarrow{\mathbf{d}}_i$ of the LSTM is given to a feed-forward neural network with 1 hidden layer, which outputs the corresponding probability vector $\mathbf{a}_i$.

Label sequence optimization layer

The label sequence optimization layer takes the sequence of probability vectors $\mathbf{a}_{1:n}$ from the label-prediction layer as input and outputs a sequence of labels $y_{1:n}$, where y_i is the label assigned to the token x_i.

The simplest strategy to select the label y_i would be to choose the label that has the highest probability in $\mathbf{a}_i$, i.e., $y_i = \text{argmax}_k \mathbf{a}_i[k]$. However, this greedy approach fails to take into account the dependencies between subsequent labels. For example, it might be more likely to have a token with the PHI type STATE followed by a token with the type ZIP than any other PHI type. Even though the label-prediction layer has the capacity to capture such dependencies to a certain degree, it might be preferable to allow the model to directly learn these dependencies in the last layer.

Table 3. Overview of the i2b2 and MIMIC datasets

Statistics	i2b2	MIMIC
Vocabulary size	46 803	69 525
Number of notes	1304	1635
Number of tokens	984 723	2 945 228
Number of PHI instances	28 867	60 725
Number of PHI tokens	41 355	78 633

One way to model such dependencies is to incorporate a matrix T that contains the transition probabilities between two subsequent labels. $T[i,j]$ is the probability that a token with label i is followed by a token with the label j. The score of a label sequence $y_{1:n}$ is defined as the sum of the probabilities of individual labels and the transition probabilities:

$$s(y_{1:n}) = \sum_{i=1}^{n} \mathbf{a}_i[y_i] + \sum_{i=2}^{n} T[y_{i-1}, y_i].$$

These scores can be turned into probabilities of the label sequences by taking a softmax function over all possible label sequences. During the training phase, the objective is to maximize the log probability of the gold label sequence. In the testing phase, given an input sequence of tokens, the corresponding sequence of predicted labels is chosen as the one that maximizes the score.

EXPERIMENTS AND RESULTS

Datasets

We evaluate our two models on two datasets: i2b2 2014 and MIMIC. The i2b2 2014 dataset was released as part of the 2014 i2b2/UTHealth shared task Track 1.[29] It is the largest publicly available dataset for de-identification. Ten teams participated in this shared task, and 22 systems were submitted. As a result, we used the i2b2 2014 dataset to compare our models against state-of-the-art systems.

The MIMIC de-identification dataset was created for this work as follows: the MIMIC-III dataset[7,8,56] contains data for 61 532 intensive care unit stays over 58 976 hospital admissions for 46 520 patients and includes 2 million patient notes. In order to make the notes publicly available, a rule-based de-identification system[5,6,57] was written for the specific purpose of de-identifying patient notes in MIMIC, leveraging dataset-specific information such as patient names or addresses. The system favors recall over precision: there are virtually no false negatives, while there are numerous false positives. To create the gold standard MIMIC de-identification dataset, we selected 1635 discharge summaries, each belonging to a different patient, containing a total of 60 700 PHI instances. We then annotated the PHI instances detected by the rule-based system as true positives or false positives and found that 15% were false positives.

Table 1 introduces the PHI types used as labels for training, and Table 3 presents the sizes of the datasets. For the test set, we used the official test set for the i2b2 dataset, which is 40% of the dataset; we randomly selected 20% of the MIMIC dataset as the test set for this dataset.

Evaluation metrics

To assess the performance of the two models, we computed the precision, recall, and F1-score. Let TP be the number of true positives, FP the number of false positives, and FN the number of false negatives. Precision, recall, and F1-score are defined as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP}+\text{FP}},\ \text{recall} = \frac{\text{TP}}{\text{TP}+\text{FN}},\ \text{and}$$
$$\text{F1-score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}.$$

Intuitively, precision is the proportion of predicted PHI labels that are gold labels, recall is the proportion of gold PHI labels that are correctly predicted, and F1-score is the harmonic mean of precision and recall.

Training and hyperparameters

The model is trained using stochastic gradient descent, updating all parameters, i.e., token embeddings, character embeddings, parameters of bidirectional LSTMs, and transition probabilities, at each gradient step. For regularization, dropout is applied to the character-enhanced token embeddings before the label-prediction layer. Training the model takes approximately 2 days on an Nvidia Titan X graphics processing unit for the i2b2 dataset. The actual running time depends on the choice of hyperparameters, the weight initialization, and the size of the dataset.

Below are the choices of hyperparameters and token embeddings, optimized using a subset of the training set:

- Character embedding dimension: 25
- Character-based token-embedding LSTM dimension: 25
- Token embedding dimension: 100
- Label prediction LSTM dimension: 100
- Dropout probability: 0.5

As mentioned previously, token embeddings can be pre-trained, and during training the token mapping $\mathcal{V}_T(\cdot)$ is initialized with the pre-trained token embeddings. We tried pre-training token embeddings on the i2b2 2014 and MIMIC datasets (for MIMIC, we used the entire dataset containing 2 million notes and 800 million tokens), using word2vec and GloVe. Both of these were trained using a window size of 10, a minimum vocabulary count of 5, and 15 iterations. Additional parameters of word2vec were negative sampling and model type, which were set to 10 and skip-gram, respectively. We also experimented with the publicly available token embeddings such as GloVe (http://nlp.stanford.edu/projects/glove/) trained on Wikipedia and Gigaword 5.[58] The results were quite robust to the choice of pre-trained token embeddings. The GloVe embeddings trained on Wikipedia articles yielded slightly better results, and we chose them for the rest of this work.

Results

All results were computed using the official evaluation script from the i2b2 2014 de-identification challenge. Table 4 presents the main results, based on binary token-based precision, recall, and F1-score for HIPAA-defined PHI only. These PHI types are the most important, since only these are legally required to be removed. The results for each PHI type, dataset, and system are presented in Supplementary Appendix 1, Tables A1 and A2.

On the i2b2 dataset, our ANN model has a higher F1-score and recall than our CRF model and the best system from the i2b2 2014 de-identification challenge, the Nottingham system.[51] The only freely available off-the-shelf program for de-identification, called the MITRE Identification Scrubber Toolkit (MIST),[27] performed the worst. The outputs of our ANN and CRF models can be combined by considering a token to be PHI if it is identified as such by either model. This further increases the performance in terms of F1-score and recall. It should be noted that the Nottingham system was specifically fine-tuned for the i2b2 dataset and the i2b2 evaluation script. For example, the Nottingham system post-processes the detected PHI terms in order to match the offset of the gold PHI tokens, such as modifying “MR:6746781” to “6746782” and “MWFS” to “M,” “W,” “F,” “S.”

Table 4. Performance (%) on the PHI as defined in HIPAA

	i2b2			MIMIC		
Model	Precision	Recall	F1	Precision	Recall	F1
Nottingham	**99.000**	96.400	97.680	–	–	–
MIST	91.445	92.745	92.090	95.867	98.346	97.091
CRF	98.560	96.528	97.533	99.060	98.987	99.023
ANN	98.320	97.380	97.848	**99.208**	99.251	**99.229**
CRF + ANN	97.920	**97.835**	**97.877**	98.820	**99.398**	99.108

We evaluated the systems based on detection of PHI tokens vs. non-PHI tokens (i.e., binary HIPAA token-based evaluation). The best performance for each metric on each dataset is highlighted in bold. Nottingham is the best performing system from the 2014 i2b2/UTHealth shared task Track 1. MIST is a freely available de-identification program. CRF is the model based on conditional random fields, ANN is the model based on artificial neural networks, and CRF+ANN is the result obtained by combining the outputs of the CRF and ANN models. The tagsets used for training the CRF and ANN models are the same as in Table 1, and the configuration of MIST is presented in Supplementary Appendix 2. Note that the performance of the MIST system cannot be directly compared to that of other systems, because the tagsets used are different. The Nottingham system could not be run on the MIMIC dataset, as it is not publicly available.

On the MIMIC dataset, our ANN model also had a higher F1-score and recall than our CRF model. Interestingly, combining the outputs of our ANN and CRF models did not increase the F1-score, because precision was negatively impacted. However, recall did benefit from combining the two models. MIST was much more competitive on this dataset.

We calculated the statistical significance of the differences in precision, recall, and F1-score between the CRF and ANN models using approximate randomization with 9999 shuffles. The significance levels of the differences in precision, recall, and F1-score were 0.37, 0.02, 0.22 for the i2b2 dataset and 0.08, 0.00, 0.00 for the MIMIC dataset, respectively.

Error analysis

Figure 2 shows the binary token-based F1-scores for each PHI category. The ANN model outperformed the CRF model on all categories for both datasets, with the exception of the ID category (which mostly contains medical record numbers) in the i2b2 dataset. This is due to the fact that the CRF model uses sophisticated regular expression features that are tailored to detect ID patterns such as “38:Z8912708G.”

Another interesting difference between the ANN and CRF results was the PROFESSION category, where the ANN significantly outperformed the CRF. The reason behind this is that the embeddings of the tokens that represent a profession tend to be close in the

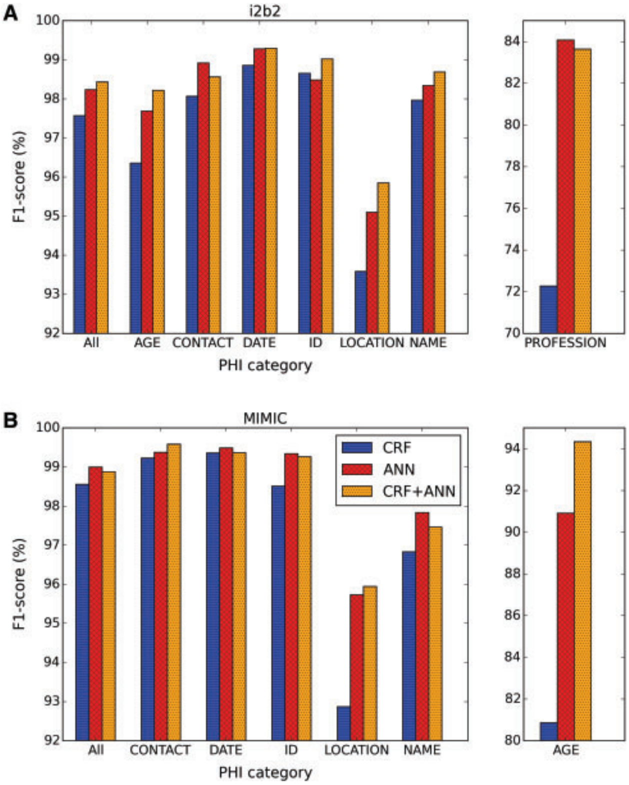

Figure 2. Binary token-based F1-scores for each PHI category. The evaluation is based on PHI types that are defined by HIPAA as well as additional types specific to each dataset. Each PHI category and the corresponding types are defined in Table 1. The "All" category refers to the F1-score micro-averaged over all PHI categories. The PROFESSION category exists only in the i2b2 dataset and was plotted separately to avoid distorting the y-axis. For the same reason, the AGE category in MIMIC was drawn separately.

token-embedding space, which allows the ANN model to generalize well. We tried assembling various gazetteers for the PROFESSION category, but all of them performed significantly worse than the ANN model.

Table 5 presents some examples of gold PHI instances correctly predicted by the ANN model that the CRF model failed to predict, and conversely. This illustrates that the ANN model efficiently copes with the diversity of contexts in which tokens appear, whereas the CRF model can only address contexts that are manually encoded as features. In other words, the ANN model's intrinsic flexibility allows it to better capture the variances in human language than the CRF model. For example, it would be challenging and time-consuming to engineer features for all possible contexts, such as "had a stroke at 80," "quit smoking in 08," "on the 29th of this month," and "his friend Epstein." The ANN model is also very robust to variations in surface forms, such as misspellings (e.g., "in teh late 60s," "Khazakhstani," "01/19/:0"), tokenizations (e.g., "Results02/20/2087," "MC # 0937884Date"), and different phrases referring to the same semantic meaning (e.g., "San Rafael Mount Hospital," "Rafael Mount," "Rafael Hospital"). Furthermore, the ANN model is able to detect many PHI instances despite not having explicit gazetteers, as examples in the LOCATION and PROFESSION categories illustrate. We conjecture that character-enhanced token embeddings contain rich enough information to effectively function as gazetteers, as tokens with similar semantics are closely located in the vector representation.[26,27,41]

On the other hand, CRF is good at capturing rarely occurring patterns that are written in highly specialized regular expression patterns (e.g., "38:Z8912708G," "53RHM") or tokens that are included in the gazetteers (e.g., "Christmas," "WPH," "rosenberg," "Motor Vehicle Body Repairer"). For example, the PHI token "Christmas" only occurs in the test set, and unless the context gives a strong indication, the ANN model cannot detect it, whereas the CRF model can, as long as it is included in the gazetteers.

Table 6 presents examples of PHI instances that are false negatives in the system that combines CRF and ANN outputs. In other words, these PHI instances are detected by neither CRF nor ANN. The sources of errors can be classified into four main categories:

- **Abbreviations**: Some PHI instances are abbreviations, which are sometimes challenging to detect, especially when they are short and ambiguous.
- **Ambiguities**: A human reader may not be able to tell whether a token is PHI. Examples include names involving common words, or numbers that could be dates or test results. Ambiguities can stem from the token itself or its context.
- **Data sparsity**: The training samples do not contain enough PHI instances similar to the ones that are missed in the test set. Also, some PHI instances are more difficult to detect than others and subsequently require more training samples.
- **Debatable annotations:** Some tokens are questionably marked as PHI instances.

Abbreviations and ambiguities are among the most challenging sources of errors to address in order to further improve the performance. We anticipate that the data sparsity issues may partly be resolved by increasing the size of the training set to contain more instances of difficult PHI types.

Effect of training set size

Figure 3 shows the impact of the training set size on the performance of the models on the MIMIC dataset. When the training set size is very limited, CRF performs slightly better than ANN, since the CRF model can leverage handcrafted features without much training data. As the training set size increases, the ANN model starts to significantly outperform the CRF model, since the parameters including the embeddings are automatically fine-tuned with more data, and therefore the features learned by the ANN model become increasingly more refined than the manually handcrafted features. As a result, combining the outputs of the CRF and ANN models increases the F1-score over the ANN model for only small training set sizes and yields a less competitive F1-score than the ANN model for bigger training set sizes.

Figure 4 details the impact of the number of labeled PHI instances in the training set on the model's performance for a given PHI type in the i2b2 dataset. As expected, PHI types with a large number of labeled PHI instances tend to be detected more accurately than rarer types. However, the correlation is far from perfect: some PHI types with a lower number of labeled instances are detected more accurately than some types with a higher number of labeled instances. This indicates that some PHI types are harder to detect than others. For example, although the PHI type "PHONE" has fewer labeled PHI instances than the type "PROFESSION" (310 vs 425 instances), the former is much more accurately detected than the latter (F1-score of 99.272 vs 86.642); this result is expected, since tokens containing a phone number are typically very similar, whereas professions can appear in many different forms.

Table 5. Examples of correctly detected PHI instances (in bold) by the ANN and CRF models for the i2b2 dataset

PHI category	ANN	CRF
AGE	Father had a stroke at **80** and died of?another stroke at age Personal data and overall health: Now **63**, despite his FH: Father: Died @ **52** from EtOH abuse (unclear exact etiology) Tobacco: smoked from age 7 to **15**, has not smoked since 15.	HPI: **53**RHM who going to bed Wednesday was in usoh, but Tobacco: Quit at **38** y/o; ETOH: 1-2 beers/week;
CONTACT	History of Present Illness **86F** reports worsening b/l leg pain. by phone, Dr. Ivan Guy. Call w/ questions **86383**. Keith Gilbert, H/O paroxysmal afib VNA **171-311-7974** ======= Medications	
DATE	During his **May** hospitalization he had dysphagia Social history: divorced, quit smoking in **08**, sober x 10 yrs, She is to see him on the **29th** of this month at 1:00 p.m. He did have a renal biopsy in teh late **60s** adn thus will look for results, Results**02/20/2087** NA 135, K 3.2 (L), CL 96 (L), CO2 30.6, BUN 1 Jose Church, M.D. /ray DD: 01/18/20 DT: **01/19/:0** DV: 01/18/20	She is looking forward to a good **Christmas**. She is here
ID	placed 3/23 for bradycardia. P/G model # **5435**, serial # 4712198, Consult NotePt: Ulysses Ogrady MC # **0937884**Date: 10/07/69	DD:05/05/2095 DT:05/05/2095 **WK:65255 :4653** NO GROWTH TO DATE Specimen: **38:Z8912708G**
LOCATION	Works in programming at **Audiovox.** Formerly at BrightPoint. He has remote travel hx to the **Rockefeller Centre**, more recent global History of Present Illness: Pt is a 59 yo **Khazakhstani** male, with who was admitted to **San Rafael Mount Hospital** following a syncopal nauseas and was brought to **Rafael Mount** ED. Five weeks ago prior Anemia: On admission to **Rafael Hospital**, Hb/Hct: 11.6/35.5.	2nd set biomarkers (**WPH**): Creatine Kinase Isoenzymes Hospitalized 2115 **TCH** for ROMI 2120 TCH new
NAME	ATCH: 655-75-45 Dear Harry and **Yair**: My thanks for your kind Patient lives in Flint with his friend **Epstein**. He has 3 children. Health care proxy-Yes, son (**West**) Allergies DUTASTERIDE - cough,	Lab Tests **Amador**: the lab results show good levels of 05/10/2066 - 04/15/2068 ACT: **rosenberg** 128 Williams Ct **M OSCAR, JOHNNY** Hyderabad, WI
PROFESSION	Social history: Married, **glazier**, 3 grown adult children Has VNA. Former civil engineer, **supervisor**, consultant. He was formerly self-employed as a **CPA** and would often travel Communications senior manager, **marketing,** worked for Brinker and Concrete Finisher (25yrs). He is a **veteran**. Former tobacco user, works part time in **securities**.	He is retried **Motor Vehicle Body Repairer**.

The examples in the ANN column are only predicted by the ANN model and not by the CRF model, and conversely. Typographical errors are from the original text.

Ablation analysis

In order to quantify the importance of various elements of the ANN model, we tried four variations of the model, eliminating different elements one at a time. Figure 5 presents the results of the ablation tests. Removing either the label sequence optimization layer, pre-trained token embeddings, or token embeddings slightly decreased the performance. Surprisingly, the ANN performed pretty well with only character embeddings and without token embeddings, and eliminating character embeddings was more detrimental than eliminating token embeddings. This suggests that character-based token embeddings may be capturing not only sub-token level features, but also the semantics of the tokens themselves.

CONCLUSION

We proposed the first system based on ANN for patient note de-identification. It outperforms state-of-the-art systems based on CRF on two datasets, while requiring no handcrafted features. Utilizing both token and character embeddings, the system can automatically learn effective features from data by fine-tuning the parameters. It jointly learns the parameters for the embeddings, the bidirectional LSTMs, and the label sequence optimization, and can make use of token embeddings pre-trained on large unlabeled datasets. Quantitative and qualitative analysis of the ANN and CRF models indicates that the ANN model better incorporates context and is more flexible to variations inherent in human languages than the CRF model.

From the viewpoint of deploying an off-the-shelf de-identification system, our results in Table 4 demonstrate recall on the MIMIC discharge summaries of over 99%, which is quite encouraging. Figure 2, however, shows that the F1-score on the NAME category, probably the most sensitive PHI type, falls just below 98% for the ANN model. We anticipate that adding gazetteer features based on the local institution's patient and staff census should improve this result, which will be explored in future work.

CONTRIBUTORS

FD and JYL contributed equally to this work. They designed and implemented the CRF and ANN models, annotated the MIMIC de-identification dataset, evaluated the systems' performance, created the figures, and wrote the paper. OU and PS formulated the original problem, provided direction and guidance, and gave helpful feedback on the paper.

Table 6. Examples of PHI instances undetected by CRF + ANN (i.e., undetected by both CRF and ANN) for the i2b2 dataset

PHI categories	PHI type	Examples	Reason	FN	Support
AGE	AGE	A **seventy-one**-year-old woman with multiple medical	S	19	790
		died of sudden death in their **82nd** year. Brother had SCD at **66**.	S		
		smoked from age 7 to 15, has not smoked since **15**.	S		
		d 80s?cause, MGF d**90** age, MGM d**73** CVAM d 73	S		
		stomach Ca, OA, obeseF d **84** multi-infarct dementiaS b66	S		
CONTACT	PHONE	Wheatland Manor: 154-734-1487, x**557** (4th floor)	S	1	410
	FAX	Phone: (091)920-5569 Fax: **(251)628-xxxx**	S	3	6
	EMAIL	E-Mail: **iparedes@oachosp.org**	S	3	3
DATE	DATE	PARONYCHIAL INFECTION: LEFT HAND **78**||Ectopic pregnancy: **74**	Am	60	12534
		alb 4.2|fe 50, tibc 204, ferritin 878 **8/27**|inr 1.1|pth 115 8/27	Am		
		Prior HDL 19.**8/67** TC 170, TG 162, H40, L98	Am		
		Referral submitted to GI**6/65**: saw GI - going for scope to eval pancreas	Am		
		DMSon b93D b94 GC due**22**D Fran b03 Abn	S		
		last seen in clinic in ||**11-70** after which time she left for	S		
ID	IDNUM	Influenza vaccine || Received 11/95 **MLL**||||	Am	9	382
		disp #100 order number **38/48**||ALLERGY||NKDA	S		
	MEDICALRECORD	Patient: Vincent Ware (71417347 **2Y**)	S	1	732
	DEVICE	Interrogation today of his Medtronic Kappa **QQ 626** pacemaker	S	4	12
LOCATION	STREET	–		0	416
	CITY	Oriented to "LCC" in "**Galena**," "March 2095." Speech fluent in Dutch.	S	8	344
	ZIP	–		0	144
	STATE	BP has been well-controlled in **VA**, usually in the 128 systolic range.	Ab/Am	9	205
	COUNTRY	is here with her husband who is translating from **columbian**.	S	13	130
	LOCATION-OTHER	travel hx to the Rockefeller Centre, more recent **global** travel	D	12	20
		and has infrequently visited **Storting** and **Acropolis**.	S		
	ORGANIZATION	diabetes diet - he enjoys a blueberry muffin from **RR Donnelley** daily.	S	42	147
		his level of fatigue. He continues to go to **the library** daily. He continues	D		
	HOSPITAL	were placed at Pomeroy Care Center (Big Rapids, **AC**) and also he	Ab/Am	44	1595
		Medication List for QUICK,ISABELLE Y 6557545 (**ATCH**) 52 F	Ab		
		2. DM, stable, Glyburide increased at **MS**. Dietary rec's||reviewed.	Ab/Am		
NAME	PATIENT	DMSon b93D b94 GC due22D **Fran** b03 Abn pap24 Nephropathy 3/25	Am	6	1450
		(HCP, daughter) 625-248-3647; **Flowers** (son) 705-690-8475	Am		
		Patient Name: JIMENEZ,YOUSSEF **I** [0554733(LCH)]	Ab/Am		
	DOCTOR	Insley/Endocrinology - End 6||**Lane**/Neurology - NEU 265	Am	35	3297
		Script: Amt: 30 Refill: 3 Date: 03/11/2074: **um**	Am		
		If the latter, will change it.||**O**||||Plasma Sodium 138	Ab/Am		
	USERNAME	–		0	92
PROFESSION	PROFESSION	however he would like to try to **intern**, when he feels up to it.	D	69	340
		Patient lives in Lake Pocotopaug with wife. **Justice of the peace**.	S		
		On disability. Volunteers - **animal rescue**. No current or previous tobacco	S		
		Social History||**NP** in Laplace - waiting for researcher job.	Ab/Am		
		He has continued actively **managing production** and is planning a trip to Italy next	S/D		

Each row presents one or two false negatives (marked in bold and underlined). The "Reason" column specifies what we believe is the main factor that caused CRF + ANN to fail to detect tokens as PHI instances. Ab: abbreviation; Am: ambiguity; D: debatable annotation; S: data sparsity. The "FN" column indicates how many tokens of a given PHI type are false negatives. The "Support" column indicates the number of tokens of a given PHI type in the test set.

FUNDING

The project was supported by Philips Research. The content is solely the responsibility of the authors and does not necessarily represent the official views of Philips Research.

COMPETING INTERESTS

The authors have no competing interests to declare.

ACKNOWLEDGMENTS

We warmly thank Michele Filannino, Alistair Johnson, Li-wei Lehman, Roger Mark, and Tom Pollard for their helpful suggestions and technical assistance.

SUPPLEMENTARY MATERIAL

Supplementary material is available at *Journal of the American Medical Informatics Association* online.

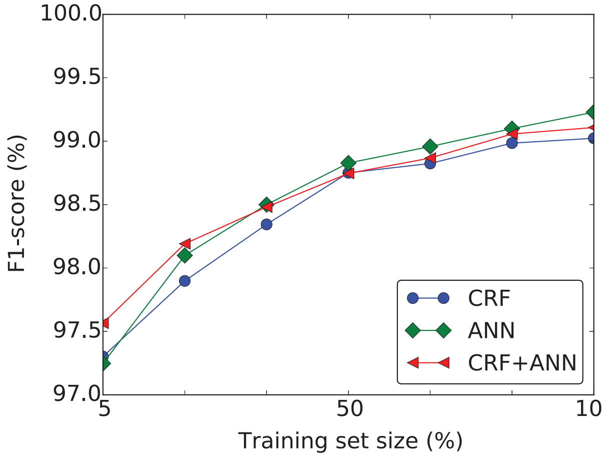

Figure 3. Impact of the training set size on the binary HIPAA token-based F1-scores on the MIMIC dataset. The 100% training set size refers to using all of the dataset minus the test set, which amounts to 2 046 488 tokens and 42 531 PHI instances. As expected, both CRF and ANN models benefit from having more training samples.

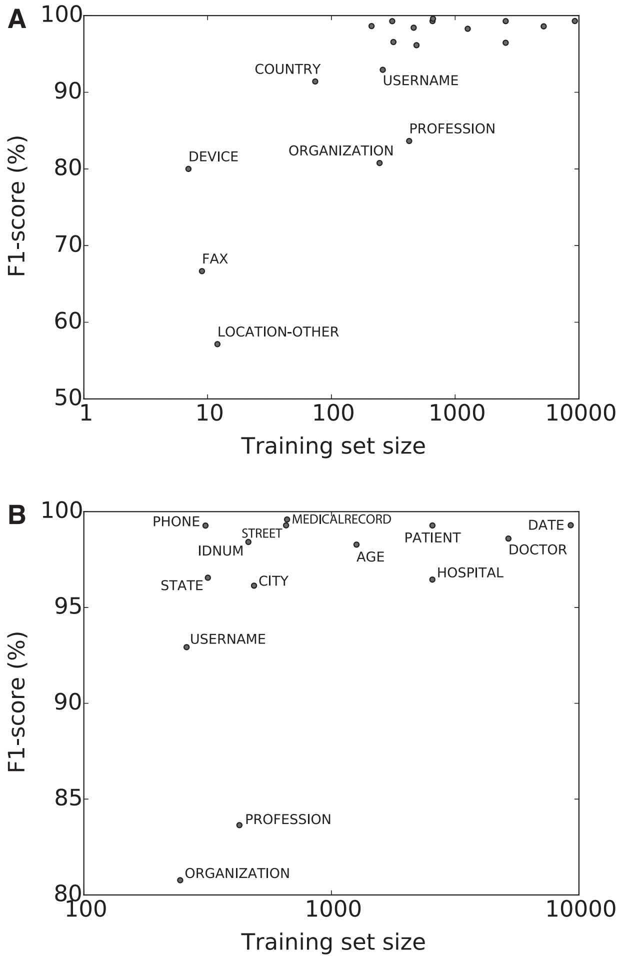

Figure 4. Impact of the number of labeled PHI instances in the training set on the model's performance for each PHI type in the i2b2 dataset. Figure (**A**) presents all PHI types, and Figure (**B**) focuses on the most commonly occurring PHI types. Having more PHI instances in the training set helps increase F1-score, but some PHI types are harder to detect than others.

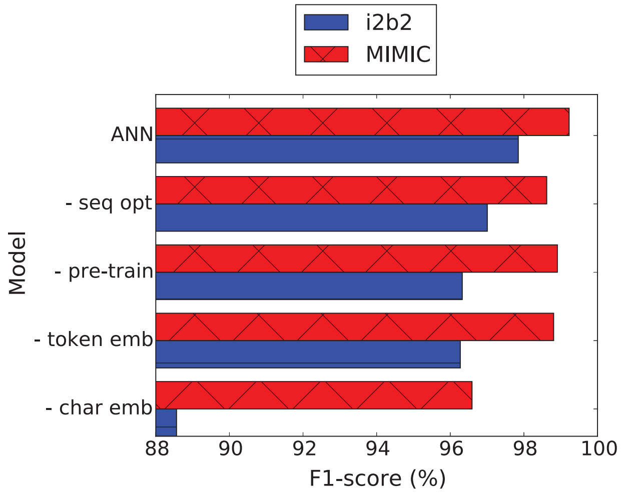

Figure 5. Ablation test performance based on binary HIPAA token-based evaluation. ANN is the model based on artificial neural networks. "– seq opt" is the ANN model without the label sequence optimization layer. "– pre-train" is the ANN model where token embeddings are initialized with random values instead of pre-trained embeddings. "– token emb" is the ANN model using only character-based token embeddings, without token embeddings. "– character emb" is the ANN model using only token embeddings, without character-based token embeddings.

REFERENCES

1. DesRoches CM, Worzala C, Bates S. Some hospitals are falling behind in meeting "meaningful use" criteria and could be vulnerable to penalties in 2015. *Health Affairs*. 2013;32:1355–60.
2. Wright A, Henkin S, Feblowitz, et al. Early results of the meaningful use program for electronic health records. *New Engl J Med*. 2013;368:779–80.
3. Office for Civil Rights H. Standards for privacy of individually identifiable health information. *Final rule*. Federal Register. 2002;67:53181.
4. Neamatullah I, Douglass MM, Li-wei HL, et al. Automated de-identification of free-text medical records. *BMC Med Inform Decis Mak*. 2008;8:1.
5. Douglass M, Clifford G, Reisner A, et al. De-identification algorithm for free-text nursing notes. *Comput Cardiol*. 2005:331–34.
6. Douglas M, Clifford G, Reisner A, et al. Computer-assisted de-identification of free text in the MIMIC II database. *Comput Cardiol*. 2004:341–44.
7. Goldberger AL, Amaral LA, Glass L, et al. Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. *Circulation*. 2000;101:e215–20.
8. Saeed M, Villarroel M, Reisner AT, et al. Multiparameter intelligent monitoring in intensive care II (MIMIC-II): A public-access intensive care unit database. *Crit Care Med*. 2011;39:952.
9. Lingren T, Deleger L, Molnar K, et al. Pre-annotating clinical notes and clinical trial announcements for gold standard corpus development: Evaluating the impact on annotation speed and potential bias. *Proc 2012 IEEE 2nd Conf Healthc Informatics, Imaging Syst Biol HISB 2012*. 2012:108. doi:10.1109/HISB.2012.33.
10. South BR, Mowery D, Suo Y, et al. Evaluating the effects of machine pre-annotation and an interactive annotation interface on manual de-identification of clinical text. *J Biomed Inform*. 2014;50: 162–72.
11. Hanauer D, Aberdeen J, Bayer S, et al. Bootstrapping a de-identification system for narrative patient records: Cost-performance tradeoffs. *Int J Med Inform*. 2013;82(9):821–31.
12. Gobbel GT, Garvin J, Reeves R, et al. Assisted annotation of medical free text using RapTAT. *J Am Med Inform Assoc*. 2014;21(5):833–41.

13. Chiticariu L, Li Y, Reiss FR. Rule-based information extraction is dead! Long live rule-based information extraction systems! *EMNLP*. 2013:827–32.
14. Berman JJ. Concept-match medical data scrubbing: how pathology text can be used in research. Arch Pathol Lab Med. 2003;127:680–86.
15. Beckwith BA, Mahaadevan R, Balis UJ, et al. Development and evaluation of an open source software tool for deidentification of pathology reports. *BMC Med Inform Decis Mak*. 2006;6:1.
16. Fielstein E, Brown S, Speroff T. Algorithmic de-identification of VA medical exam text for HIPAA privacy compliance: preliminary findings. *Medinfo*. 2004;1590.
17. Friedlin FJ, McDonald CJ. A software tool for removing patient identifying information from clinical documents. *J Am Med Inform Assoc*. 2008;15:601–10.
18. Gupta D, Saul M, Gilbertson J. Evaluation of a deidentification (De-Id) software engine to share pathology reports and clinical documents for research. *Am J Clin Pathol*. 2004;121:176–86.
19. Morrison FP, Li L, Lai AM, et al. Repurposing the clinical record: Can an existing natural language processing system de-identify clinical notes? *J Am Med Inform Assoc. 2009*;16:37–39.
20. Ruch P, Baud RH, Rassinoux A-M, et al. Medical document anonymization with a semantic lexicon. *Proc AMIA Symp*. 2000:729–33.
21. Sweeney L. Replacing personally identifying information in medical records, the Scrub system. *Proc AMIA Annual Fall Symp*. 1996:333–7.
22. Thomas SM, Mamlin B, Schadow G, et al. A successful technique for removing names in pathology reports using an augmented search and replace method. *Proc AMIA Symp*. 2002:777–81.
23. Szarvas G, Farkas R, Kocsor A. A multilingual named entity recognition system using boosting and c4.5 decision tree learning algorithms. *Discovery Sci*. 2006;4265:267–78.
24. Guo Y, Gaizauskas R, Roberts I, et al. Identifying personal health information using support vector machines. *I2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*. 2006:10–11.
25. Uzuner Ö, Sibanda TC, Luo Y, et al. P. A de-identifier for medical discharge summaries. *Artif Intell Med*. 2008;42:13–35.
26. Hara K. Applying a SVM based chunker and a text classifier to the deid challenge. *I2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*. 2006:10–1.
27. Aberdeen J, Bayer S, Yeniterzi R, et al. The MITRE Identification Scrubber Toolkit: Design, training, and assessment. Int J Med Inform. 2010;79:849–59.
28. Meystre SM, Friedlin FJ, South BR, et al. Automatic de-identification of textual documents in the electronic health record: a review of recent research. *BMC Med Res Methodol*. 2010;10:1.
29. Stubbs A, Kotfila C, Uzuner Ö. Automated systems for the de-identification of longitudinal clinical narratives: overview of 2014 i2b2/UTHealth shared task track 1. *J Biomed Inform*. 2015;58:S11–19.
30. Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. *Adv Neural Inf Process Syst*, 2013:3111–19.
31. Collobert R, Weston J, Bottou L, et al. Natural language processing (almost) from scratch. *J Mach Learning Res*. 2011;12:2493–537.
32. Pennington J, Socher R, Manning CD. GloVe: Global vectors for word representation. Proc Empiricial Methods Natural Language Processing (EMNLP 2014). 2014;12:1532–43.
33. Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model. *Interspeech*. 2010:3.
34. Socher R, Perelygin A, Wu JY, et al. Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2013;1642:18–21.
35. Kim Y. Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. 2014:1746–51.
36. Blunsom P, Grefenstette E, Kalchbrenner N, et al. A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. 2014;1:655–65.
37. Lee JY, Dernoncourt F. Sequential short-text classification with recurrent and convolutional neural networks. *Human Language Technologies 2016: The Conference of the North American Chapter of the Association for Computational Linguistics*. 2016:515–20.
38. Weston J, Bordes A, Chopra S, et al. Towards AI-complete question answering: a set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*. 2015.
39. Wang D, Nyberg E. A long short-term memory model for answer sentence selection in question answering. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (vol. 2: short papers) [Internet]*. Beijing, China: Association for Computational Linguistics; 2015:707–12.
40. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*. 2014.
41. Tamura A, Watanabe T, Sumita E. Recurrent neural networks for word alignment model. *ACL (1)*. 2014;52:1470–80.
42. Sundermeyer M, Alkhouli T, Wuebker J, et al. Translation modeling with bidirectional recurrent neural networks. *EMNLP*. 2014:14–25.
43. Lample G, Ballesteros M, Subramanian S, et al. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*. 2016.
44. Labeau M, Löser K, Allauzen A. Non-lexical neural architecture for fine-grained POS tagging. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing [Internet]*. Lisbon, Portugal: Association for Computational Linguistics; 2015;14:232–37.
45. Kim Y, Jernite Y, Sontag D, et al. Character-aware neural language models. *arXiv preprint arXiv:1508.06615*. 2015.
46. Wu Y, Jiang M, Lei J, et al. Named entity recognition in Chinese clinical text using deep neural network. *Stud Health Technol Inform*. 2015;216:624–28.
47. Li P, Huang H. UTA DLNLP at SemEval-2016 Task 12: deep learning based natural language processing system for clinical information identification from clinical notes and pathology reports. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, CA: Association for Computational Linguistics; 2016;10:1268–73.
48. Fries JA. Brundlefly at SemEval-2016 Task 12: recurrent neural networks vs. joint inference for clinical temporal information extraction. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, CA: Association for Computational Linguistics; 2016:1274–9.
49. Zhang C. DeepDive: a data management system for automatic knowledge base construction. *Thesis*. 2015;53:1689–99.
50. Manning CD, Bauer J, Finkel J, et al. The Stanford CoreNLP Natural Language Processing Toolkit. *Proc 52nd Annu Meet Assoc Comput Linguist Syst Demonstr*. 2014;52:55–60.
51. Yang H, Garibaldi JM. Automatic detection of protected health information from clinic narratives. *J Biomed Inform*. 2015;58:S30–38.
52. Filannino M, Brown G, Nenadic G. ManTIME: temporal expression identification and normalization in the TempEval-3 challenge. *CoRR*. 2013;abs/1304.7(2005):5.
53. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput*. 1997;9:1735–80.
54. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*. 2013.
55. Mikolov T, Yih W-t, Zweig G. Linguistic regularities in continuous space word representations. *HLT-NAACL*. 2013;13:746–51.
56. Johnson AEW, Pollard TJ, Shen L, et al. MIMIC-III, a freely accessible critical care database. Scientific Data. 2016; 3:160035.
57. Douglass M. *Computer-assisted De-identification of Free-Text Nursing Notes [Master's thesis]*. Cambridge, United States: Massachusetts Institute of Technology; 2005.
58. Parker R, Graff D, Kong J, et al. *English Gigaword Fifth Edition, Linguistic Data Consortium. Technical Report*. Philadelphia: Linguistic Data Consortium; 2011.