Anonymize Sensitive Information in Medical Health Records

Bhavna Saluja, Gaurav Kumar, Srinivas Suri, Ipek Onur
{bsaluja,gauku,surisr,ionur}@seas.upenn.edu,

1 Abstract

Due to privacy constraints, clinical records with protected health information (PHI) cannot be directly shared. De-identification, i.e., the exhaustive removal, or replacement, of all mentioned PHI phrases has to be performed before making the clinical records available outside of hospitals. We have tried to identify PHI on medical records written in Spanish language. As part of simple random baseline, we analysed the data set and built a rule based system to identify PHI tags. This gave us F1 score of 0.506 on test data. In the published baseline part, we gathered various token-level features and built a LinearSVC model which gave us F1 score of 0.861 on test data. In the extension 1, we added dictionary lookup features for Spanish names and locations in addition to token-level features and improved our rule based parser. We found the top performing classifiers and used majority rule for our predictions. This method gave us a lower F1 score compared to our published baseline score.

2 Introduction

For our project, we have decided to work on implementing a robust algorithm to capture sensitive information in medical records in order to facilitate their anonymization. We used the dataset provided as part of the MEDDOCAN competition [4] and we will be submitting our model to the competition.

2.1 What is PHI?

As mentioned on the task page [4], clinical records with protected health information (PHI) cannot be directly shared "as is", due to privacy constraints, making it particularly cumbersome to carry out NLP research in the medical domain. A necessary precondition for accessing clinical records outside of hospitals is their de-identification, i.e., the exhaustive removal, or replacement, of all mentioned PHI phrases.

PHI stands for Protected Health Information and is any information in a medical record that can be used to identify an individual, and that was created, used, or disclosed in the course of providing a health care service, such as a diagnosis or treatment. In other words,

PHI is personally identifiable information in medical records, including conversations between doctors and nurses about treatment. PHI also includes billing information and any patient-identifiable information in a health insurance company's computer system. Some information that can be considered PHI are Names, Surnames, Addresses, Hospitals, Professions, Different types of locations (provinces, cities, towns,...), Billing information, Email, Phone records.

2.2 Formal definition:

1. Entity recognition on the data: The goal is to match exactly the beginning and end locations of each PHI entity tag, as well as detecting correctly the annotation type. After looking at training data, we see a good amount of examples for this task.
2. Detect sensitive spans: The goal is to identify and be able to obfuscate or mask sensitive data, regardless the actual type of entity or the correct offset identification of multi-token sensitive phrase mentions. This is a comparatively difficult task because of lack of examples in training data.

2.3 An example of the task:

In this section, we have defined a figure[1] that shows an example of such a task. In the figure[1], we can see that the sensitive data like Name, Hospital Name and SSN Id are intially visible and are masked after applying any approach that masks sensitive data. Formally, A NLP approach identified all the sensitive tags using NER approach or deep nets or n-gram models and so on ... It identified such words and masked them before releasing it to researchers.

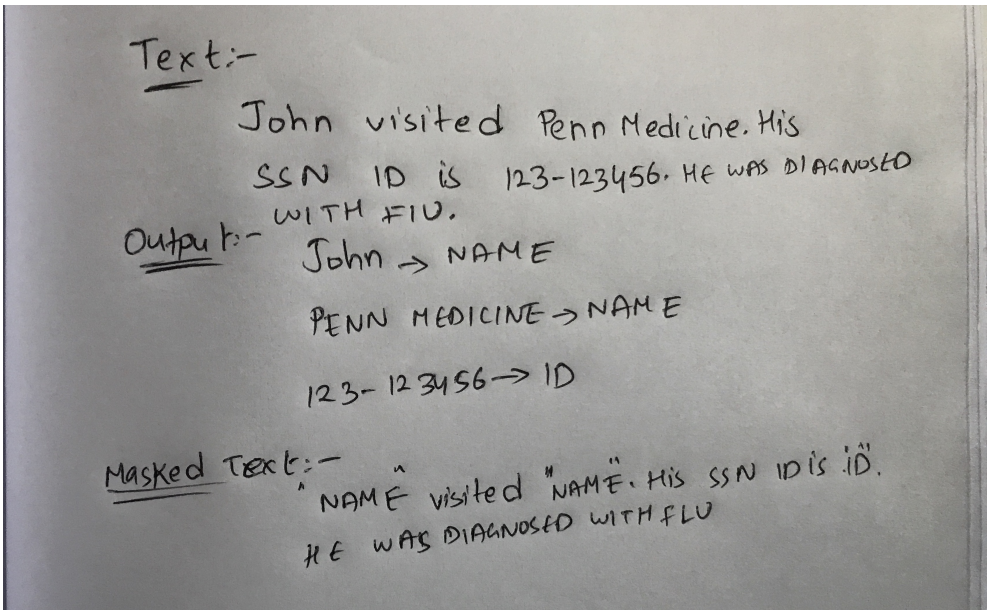

Figure 1: A prototype of the Job Information dialog

2.4 Why we choose this topic:

A necessary precondition for accessing clinical records outside of hospitals is their de-identification, i.e., the exhaustive removal, or replacement, of all mentioned PHI phrases. The practical relevance of anonymization or de-identification of clinical texts motivated us to choose this topic. Another reason why we choose this topic is because the task is specifically devoted to the anonymization of medical documents in Spanish. It is in contrast to the i2b2 effort which deeply influenced the clinical NLP community worldwide was focused on documents in English and covering characteristics of US-healthcare data providers.

3 Literature Review

We have presented below the brief descriptions of the paper we have read. There are 5 papers in total and we have used the paper[10] for our baseline implementation.

3.1 A Deep Learning Architecture for De-identification of Patient Notes: Implementation and Evaluation[7]

In this paper, the authors have presented a deep learning architecture that uses bi-directional long short-term memory networks (Bi-LSTMs) with variational dropouts and deep contextualized word embeddings while also using components such as traditional word embeddings (Glove), character LSTM embeddings and conditional random fields. Their architecture can be broken down into four distinct layers: pre-processing, embeddings, Bi-LSTM and CRF classifier. In the pre-processing layer, they break down the input document into sentences, tokens, and characters. They then use NLTK to generate a part-of-speech (POS) tag for each token. In embedding layer, for each token they concatenate the GloVe (300 dimensions) and ELMo (1024 dimensions) representations to produce a single 1324-dimensional word input vector. The concatenated word vector is then further concatenated with the character embedding vector (50 dimensions), POS one-hot encoded vector (20 dimensions), and the casing embedded vector to produce single 1394-dimensional input vector that is fed to Bi-LSTM layer. Their Bi-LSTM layer consists of 100 hidden units, uses variational dropout and have a dropout probability of 0.5. Their approach in CRF layer is identical to the Bi-LSTM-CRF model by Huang et al (Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," CoRR, 2015).

The two main data sets that they used to evaluate their architecture are the 2014 i2b2 de-identification challenge data set and the nursing notes corpus. On the task of binary F1 scores, their architecture performs similarly to the best scores achieved by other architectures with them having a slight edge on precision metrics. They have achieved precision of 0.9830, recall of 0.9737, and binary F1 score of 0.9783. For the nursing corpora, they managed to best the scores achieved by the deidentify system while also achieving a binary F1 score of over 0.812. For majority of HIPAA-PHI categories, their system performed better than system by Yang et al.

3.2 Anonymization of General Practioner Medical Records[3]

The paper aims to develop techniques and methods for semi-automated anonymization of medical record information. Some anonymization challenges, including linguistic issues (e.g. spelling and ambiguity) and determining which parts of the data that is sensitive are also discussed. The paper proposes methods like utilization of database structure, dictionaries, heuristics and natural language processing for anonymizing patient records in general. The dataset is in Norwegian language which means the task is even more challenging since it is different than English in terms of linguistic aspects. Major challenges which are posited are the differences in identity markers (e.g. Dr. and Mrs.) and hyphenation patterns in Norwegian, unstructured text, no strictly enforced guidelines for how the data should be encoded.

The goal of this paper is to anonymize general practioner data - both structural and free-text. A 6-step anonymization approach is being proposed-

A. Dictionaries - Words and numbers found in structural data can be extracted into local dictionaries with corresponding type (e.g. of patient names, social security numbers, postal codes, health institution names, health personnel names and locations).

B. Exact Match and Tag - In this step, the focus is on processing the unigram created from the free-text notes. Based on all the local and external dictionaries, a combined dictionary is created where one can look up words and get their corresponding type(s). In order to tag non-textual symbols such, e.g. dates, phone numbers and social security numbers, regular expressions are used.

C. Approximate Match and Tag - Patient records may contain erroneously spelled words, and in many cases they might be only slightly incorrectly spelled. Levenstein distance is used; by going through untagged words in the unigram and allowing an edit distance of 1, candidate misspellings can be found by relating them to the combined dictionary.

D1. Resolve Tagged Words - The words in the unigram should be replaced only if it is an identifying name (e.g. person name). When words have multiple type tags, they have to be replaced or removed if one or more of the tags are of the identifying type.

D2. Handle Untagged Words - Untagged words in the unigram needs to be investigated manually by the local clinician for tagging, the result of this investigation could be additional entries for the local or external dictionaries.

E. Final Result - The final result contains patient record text with identifying entities replaced by pseudonyms. In order to ensure an acceptable level of anonymization the result must be validated according to the requirements that motivated the anonymization in the first place.

3.3 State-of-the-art Anonymization of Medical Records Using an Iterative Machine Learning Framework[6]

The authors have developed a de-identification model that can successfully remove personal health information (PHI) from discharge records to make them conform to the guidelines of the HIPAA.

Feature set Built: Authors have used feature set of 5 different categories described below: Word level features: Word level features like Word Capitalization, punctuation marks,

digits, Roman/Arabic numbers and other common word level feature of phone numbers, fax, age and ID's etc Frequency Information: Gathered frequency of tokens from the Internet. Used features like frequency of the token, the ratio of the token's capitalized and lowercase occurrences, the ratio of capitalized and sentence beginning frequencies of the token. Offline Dictionaries: 5 different dictionaries collected from the Internet. Contextual Information: Sentence position, closest Heading to the section and several other features. Phrasal Information: a forecasted class of several preceding words and common suffixes etc.

Authors have trained 3 different classifiers that used 3 different contextual features and used a voting based mechanism to decide if the word belonged to N.E.R. If any 2 classifiers have predicted the same label, the word is assigned that tag otherwise it's not considered NER.

Authors have used an iterative approach in the following way: In the first iteration, they have collected several named entities from the headings and used these entities as an extra dictionary. They train the classifiers and collected all the texts tagged as entities from Step 1. They used all the texts tagged as entities and added this dictionary set as an extra dictionary for step 2.

3.4 Anonymizing and Sharing Medical Text Records[8]

There are three categories in medical texts: (i)explicit identifiers (EID) which includes patient name, phone number, and social security number, which can be used to directly identify an individual (ii)quasi-identifier (QID) which includes date of birth, admission/discharge date, hospital, and zip code, (iii)Health and medical details (HMDs) such as symptoms, test results, disease, and medications.

To extract these categories, a three step approach will be used. Step 1: The feature extractor: It will split the document into terms. Each term will include local features such as term length, part-of-speech, etc; global features such as the position of the term in the document. (e.g., header, body text, heading, etc.); external features regarding such as whether the term belongs to a proper noun list, belongs to a medical concept lexicon, etc

Step 2: Base classifiers: Use multiple classifiers independent of each other from the features that are extracted. Goal of the classifiers are to match the terms to EID, QID, HMD, or irrelevant categories

Step 3: Combine the results of independent classifiers to get final tags of the words

3.5 Identifying Personal Health Using Support Vector Machines[10]

The authors have developed a SVM model that can successfully remove personal health information (PHI) from discharge records to make them conform to the guidelines of the HIPAA.

The authors have experimented using SVM's to recognise NER data. Authors have built feature set using various features and trained svm on them. They have used token level features like orthographic features, length, pos, kind etc. Additionally, they have used features like date, id, phone number etc. They have used ANNIE Web API to identify hospitals, people or locations etc. The dataset of this paper's are in english and our dataset is in Spanish. Therefore, we cannot use ANNIE directly as the language is different. However,

we have implemented phone number feature. We are tokenizing date as is a number column and there are no ID's in our dataset. Therefore, we could not copy the features as it from the paper, however we tried to reciprocate the features as much as we could trying to mimic the paper.

Authors have trained their dataset on SVM. However, they have not mentioned about their default parameters. SVM kernel runs in $O(n^2)$ where n is the number of rows because of the kernel matrix computation. We used LinearSVC() which scales in linear time. This provides with several advantages over the linear SVM. First of all, it provides us an opportunity to scale the feature set for future milestones. Second, we haven't deviated from using the svm as linear svc is very similar to linear svm without computing the kernel matrix and it runs in linear time optimization.

Authors have achieved a F1-Score of 96%. However, since their dataset is different, it's difficult to compare their F-Scores with ours.

We have chosen paper 5 for implementation. There are several reasons why we went ahead with paper 5:

- From an NLP perspective, this paper forms a very good baseline paper to understand how researchers approach the problem of NER for PHI. As the feature set isn't exhaustive like some other papers, where they had almost 140 features! This paper has the practicality to be implemented for this project.

- They have achieved an f-score of 96%. Although their data was in english, their implementation looks promising because of their results.

- They have also used non machine learning features like ANNIE, JAPE grammar rules etc. which broaden our scope of implementation and experimentation. While implementing these features for Spanish language doesn't make sense, this paper forms our inspiration and provided us with a roadmap of using Spanish name dictionaries, Spanish location dictionaries for future extensions.

- There were few papers which didn't implement any machine learning at all. This paper used SVM for their implementation. This provides us with a way to extend our MS3 implementation for the next milestone by implementing more features or other type of classifiers like Neural Nets/LSTMs etc.

4 Data Preparation

We created dictionary representations of train, dev and test dataset provided as part of MEDDOCAN task. The clinical records are provided as text files. We represented each word as $(word, NER\_Tag, start\_index, Spanish\_POSTAG)$. For each clinical record, we store a list of sentences. Each sentence is further a list containing tuple representation of words in it. The records are then stored as (key, value) pairs in a dictionary and dumped as pickle files. These pickles are being used as input to all our models.

5 Experimental Design

5.1 Dataset Description:

The data is provided as part of the MEDDOCAN competition [4]. Data is in two different formats:Brat and XML. We decided to use Brat format. Figures 2 and 3 display an example input text file and output label.

```
Nombre:  Patricia.
Apellidos: Gonzalez Perez.
NHC: 7846523.
NASS: 98 91329278 42.
Domicilio: Calle Gral. Rey, 2.
Localidad/ Provincia: Ciudad Real.
CP: 13001.
Datos asistenciales.
Fecha de nacimiento: 29/08/1984.
País: España.
Edad: 33 años Sexo: M.
Fecha de Ingreso: 13/05/2017.
Servicio: Nefrologia.
Episodio: 5486916154.
Médico: Francisco Rivera Hernández NºCol: 13 13 45698.
Historia Actual: Mujer de 33 años, sin antecedentes de interés, que presenta un cuadro de dolor
abdominal, diarreas y síndrome constitucional, aparecido unas semanas después de su primer parto;
la gestación previa había transcurrido sin complicaciones. Es estudiada en la Sección de Aparato
Digestivo, siendo diagnosticada de EC al reunir criterios diagnósticos: 1) título elevado de
anticuerpos antitransglutaminasa; y 2) biopsia intestinal con atrofia de vellosidades en duodeno e
hiperplasia de criptas. Poco después de la clínica comentada, presenta poliartralgias de medianas
articulaciones, de predominio en carpos y tarsos, junto con parestesias y lesiones cutáneas con
micropápulas pruriginosas. La envían a Nefrología por detectar proteinuria de +++. En la
exploración física: tensión arterial 104/76 mmHg, IMC 27, mínimos edemas en miembros inferiores y
pápulas en codos y brazos. En la analítica: hemograma y coagulación normales, creatinina 0,9 mg/dl,
colesterol total 238 mg/dl, triglicéridos 104 mg/dl, proteínas totales 6,5 g/dl y albúmina 3,6 g/
dl. En el estudio inmunológico: ANA, anti-DNA, ANCAS, C3, C4, anticuerpos anticardiolipina,
anticoagulante lúpico y crioglobulinas negativos o normales. La serología frente a VHB, VHC y VIH
negativa. Los anticuerpos antitransglutaminasa tisular IgA positivos a título 800 U/ml (normal <7 U/
ml). En orina: proteínas 4,4 g/día y sedimento con 5 hematíes por campo. La ecografía renal es
normal. Ante la persistencia de la proteinuria nefrótica durante varias semanas, se hace biopsia
renal percutánea, cuyo resultado es una NM estadio 2, sin lesiones vasculares ni intersticiales.
Recibe tratamiento con dieta sin gluten y otras medidas conservadoras con estatina, aspirina a
dosis antiagregantes, enalapril y losartán, sin respuesta inicial. No obstante, a los 12 meses de
seguimiento, se aprecia de forma simultánea la negativización de los anticuerpos
antitransglutaminasa y la remisión completa de la proteinuria, como se indica en la figura 1.
Durante este tiempo no ha presentado otros datos de enfermedad sistémica, salvo aumento de los
niveles de TSH 6,6 µU/ml, con T4l de 1,1 ng/dl y anticuerpos antitiroglobulina y antimicrosomales
negativos.
Remitido por: Dr.Francisco Rivera Hernández, Sección de Nefrología Hospital General de Ciudad Real,
Ciudad Real. friverahdez@senofro.org
```

Figure 2: Example Text

```
T1  CORREO_ELECTRONICO 2704 2727    friverahdez@senofro.org
T3  HOSPITAL 2658 2689  Hospital General de Ciudad Real
T4  NOMBRE_PERSONAL_SANITARIO 2608 2634 Francisco Rivera Hernández
T5  SEXO_SUJETO_ASISTENCIA 396 401  Mujer
T6  EDAD_SUJETO_ASISTENCIA 405 412  33 años
T7  ID_TITULACION_PERSONAL_SANITARIO 366 377    13 13 45698
T8  NOMBRE_PERSONAL_SANITARIO 332 358   Francisco Rivera Hernández
T9  ID_CONTACTO_ASISTENCIAL 312 322 5486916154
T10 FECHAS 268 278  13/05/2017
T11 SEXO_SUJETO_ASISTENCIA 247 248  M
T12 EDAD_SUJETO_ASISTENCIA 233 240  33 años
T13 PAIS 219 225    España
T14 FECHAS 201 211  29/08/1984
T15 TERRITORIO 152 157  13001
T16 TERRITORIO 135 146  Ciudad Real
T17 CALLE 93 111    Calle Gral. Rey, 2
T18 ID_ASEGURAMIENTO 66 80  98 91329278 42
T19 ID_SUJETO_ASISTENCIA 51 58  7846523
T20 NOMBRE_SUJETO_ASISTENCIA 30 44  Gonzalez Perez
T21 NOMBRE_SUJETO_ASISTENCIA 9 17   Patricia
T2  TERRITORIO 2691 2702    Ciudad Real
```

Figure 3: Example Label

As our dataset is part of a competition, test data was scheduled to be released on April 29 with their gold labels scheduled to be released on May 20. Therefore, we rebuilt our

dataset by removing some of the medical records from the training and development set to create our test set. Table 1 describes the distribution of our dataset.

Type	Size	Avg Number of Sentences	Avg Number of Words
Training Set	401	20	393
Development Set	193	20	383
Test set	156	20	424

Table 1: Dataset distribution

Challenges and motivation: 1. Most of the research in medical domain is currently being done for English text, so doing the same for a low-resource language is challenging but much needed. 2. There is a need to share medical records to carry out research in the medical domain, so developing a system to automatically anonymize medical records will be of great help.
Link to the dataset is found here [5].

5.2 Evaluation Metric:

1. Entity recognition on the data: For this task, evaluation is entity-based. This task requires to compare the system output with the beginning and end locations of each PHI entity tag, as well as detecting correctly the annotation type.

2. Detect sensitive spans: For this task evaluation is span-based, by just evaluating whether spans belonging to sensitive phrases are detected correctly. This boils down to a classification of spans, where systems try to obfuscate spans that contain sensitive PHI expressions.

For both sub-tasks, the metrics used are micro-averaged precision, recall, and balanced F-score:

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$F = 2. \frac{Precision\ .\ Recall}{Precision + Recall}$$

In a classification task, the precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class). Recall in this context is defined as the number of true positives divided by the total number of

elements that actually belong to the positive class (i.e. the sum of true positives and false negatives, which are items which were not labeled as belonging to the positive class but should have been). The F1 score (also F-score or F-measure) is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score. The F1 score is the harmonic average of the precision and recall, where an F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.

The higher the F-score, the better is the performance of the model.

The F-score has been widely used in the natural language processing literature, such as the evaluation of named entity recognition and word segmentation. Many published papers mentioned in the literature review section have used F-score to evaluate the performance of their models.

5.3 Simple Baseline:

5.3.1 Preprocessing the data

Below are the steps we used for pre-processing the data:

Step 1 - Parse all the labels in the training data from all the files to identify (word,tag) tuples:

Our dataset is in Spanish language. We use the labelled data of the training set to identify the (word,tag) tuples. For example, if one of the labels in the training set contains (Ernesto, 'NOMBRE_SUJETO_ASISTENCIA') as one of the gold standard label for a file, we add (Ernesto,'NOMBRE_SUJETO_ASISTENCIA') to the dictionary D. We populate the dictionary D using the approach mentioned above for all the files.

Step 2 - Parse the text from the training data from all the files to identify the list L of (identifier,word) tuples:

We then parse all the files one by one for a second time. In each file, we search for all the keywords from the dictionary D. For each keyword found from the dictionary D, we search if ':' is the character that appears before the keyword. If we find ':' before the keyword, we label the word before ':' as an Identifier. For example, if we find 'Nombre: Ernesto' in the running text, we add ('Nombre') as an Identifier. After the end of second pass of the text files, we get all the (Identifier,word) tuples from all the files. From D, we have a mapping of (word,tag) and from the second pass of reading all the files, we have mappings of the form (Identifier, word). Using the transitive relation, we create a precomputed dictionary of (Identifier, tag).

Example:

Precomputed dictionary D looks something like: 'Nombre': 'NOMBRE_SUJETO_ASISTENCIA', ...

After reading all the files second time, We get, L = [('Nombre','Ernesto'),....] ('Nombre','Ernesto') is one of the (Identifier,word) tuple from the List L.

Step 3 - Combine the tuples from Dictionary D and List L to generate (identifier,tag) tuples that forms the basis for our baseline model:

We combine Dictionary D and the list L to generate the (identifier,tag) tuples. For example,('Nombre','Ernesto') is one of the (Identifier,word) tuple. All such tuples are stored as a dictionary on the disk and these tuples form the basis of our baseline model.

Step 4 - Creating the tags for the baseline model:

We use regex expressions to find the words of the form word1:word2 from the training/validation files. For each such tuples found, we see if the word1 is a key from the precomputed Dictionary D1. If it is a key from the Dictionary D1, we label the word2 with the tag corresponding to the word1 from Dictionary D1. This simple baseline does not involve any machine learning. It relies on two things 1. Finding the : separated words from the running text 2. Assigning the tag to the words using the precomputed Dictionary D1.

5.3.2 Results for our baseline model:

The results for the baseline model are presented in the table below:

Classifier	Precision	Recall	F1Score
Training Set	.634	..438	**.518**
Validation Set	.629	.425	.507
Test set	.628	.432	.506

Table 2: Baseline Scores

6 Published Baseline

We have built a baseline model simulating the kind of work implemented in the paper on 'Identifying Personal Health Information Using Support Vector Machines' [10]. Since our dataset is in Spanish which is a low-resource language, we couldn't implement some features because of the scarcity of data for this particular locale. We have tried to gather as many token-level features as we could find for Spanish.

Our feature set is as follows:

1. Word length
2. Whether word contains 1 digit
3. Whether word contains 2 digits
4. Whether word contains 3 digits
5. Whether word contains 5 digits
6. Whether word contains 6 digits
7. Whether word contains 7 digits
8. Whether word contains 9 digits
9. Whether there's an uppercase letter
10. Whether the word contains punctuations
11. Whether the word is Roman
12. Whether the word contains 'edad' or 'años'
13. Whether word is all uppercase
14. Whether word is all lowercase
15. Whether an apostrophe is present in the word
16. Whether a dash is present in the word
17. Whether the word contains 'fax'

For each word we look at the window [-1,0,1,2] and create feature vector including the features of these words in the context of the target word. We have picked these particular features because:
Digits: - Spanish phone numbers contain 9 digits and their format varies. It may come in 3 3 3, 3-3-3, 9, 2 3 2 2, 3 2 2 2, 3 6, 2 7. That is why we created a feature to capture all these different formats.
Upper/Lower cases To extract words that are part of names or addresses.
Edad/Anos to get values associated with age.
To distinguish between phone and fax numbers.

Experiments with features:
We have a total of 401 training documents, 193 dev documents and 156 test documents and we tokenized the files into sentences who counts are mentioned as follows -
Train sentences = 8300 (Total (401 docs))
Dev sentences = 4048 (Total (193 docs))
Test sentences = 3231 (Total (156 docs))

Our model used LinearSVC and the results are presented in table[3]. As we can see, we have achieved a significant improvement in results by adding these features. Our F1 score jumped to 86% after adding these new features. Although the authors achieved a score of 89% on their original paper. We cannot compare our results with them as our data is in Spanish while their data is in English.

Classifier	Precision	Recall	F1Score
Baseline	.628	.423	.506
LinearSVC	.891	.833	.861

Table 3: Baseline Scores

7 Extension 1

For our extension 1, we have extended the model submitted for Milestone 2 and 3.
We implemented extension 1 in three phases. In the first phase, we added dictionary lookup features.In the second phase, we enhanced the regex parser that we created for milestone 2. In the third phase, we found top 4 classifiers that performed the best and implemented a majority rule with those top 4 classifiers and the regex parser.

7.1 Word level features + Dictionary lookup features

As an extension of Milestone 3, we have added dictionary level features for Milestone 4. Milestone 3 had word level features and Milestone 4 classifiers are trained on word level features + dictionary lookup features. We have taken two dictionaries to built the feature set. Our first dictionary[2] is a dictionary of spanish names along with their mean age and frequency. Our second dictionary is a list of spanish location names[1] taken from wikipedia. Thereby, we have added dictionary level features to our feature set. Using the

two dictionaries mentioned here, we have added the following features on top of the features already mentioned. 18. Is word a Male Name? 19. Male Avg. Age 20. Male Name Avg. Frequency 21. Is word a Female Name? 22. Female Name Avg. Age 23. Female Name Avg. Frequency 24. Is Word a location? 25. Pos tag of words

7.2 Regex Parser

For Milestone 2, as our baseline, we were only considering the case of one instance of word1:word2 format per line. However, many lines contain more than one sensitive tag. In Milestone 4, we improved our dictionary and parser. However, with rule based methods, it is still very difficult to capture sensitive tags as not all sensitive tags come in word1:word2 format.

7.3 Majority rule

With our extended features, we tried 9 different classifiers. We found that the top performing 4 classifiers were 1.LinearSVC(C=1), 2. RandomForestClassifier($n_{estimators}$=25), 3.LogisticRegression() and 4.DecisionTreeClassifier(). Table 3 present their performances:

Classifier	Precision	Recall	F1Score
LinearSVC(C=1)	.889	.837	**.862**
RandomForestClassifier(n$_e$*stimators* = 25)	.888	.802	.843
LogisticRegression()	.874	.788	.829
DecisionTreeClassifier()	.835	.785	.809

Table 4: Word level Features + Dictionary Lookup features on Test Data

We then implemented a majority rule giving higher weight to the best performing LinearSVC() and lower weight to poorer performing DecisionTreeClassifier(). We also included the predictions from the rule based approach and received a lower performance compared to LinearSVC(C=1)'s performance alone. Table 6 displays the results of this majority rule approach on the test set.

Approach	Precision	Recall	F1Score
LinearSVC(C=1)	.889	.837	.862
Majority Rule	.8951	.8285	**.8606**

Table 5: Majority Rule Performance on Test Data

8 Extension 2

For our extension 2, we have built a named entity recognizer which is often also considered as a sequence tagging task. We took reference from the paper [9]. The model architecture

involves Bi-LSTM and CRF. Additionally, it makes use of fasttext word embeddings for Spanish. It also builds word embeddings using character encodings. We have used word embeddings(fasttext) concatenated with model word embeddings (char based Bi-LSTM) while training the model. We then extract contextual representation of each word in a given sentence by running Bi-LSTM on the sentence. In the end, we used CRF to decode the output and get the category of each word.

	Precision	**Recall**	**F1-Score**
Train Set	0.9858	0.9394	0.9620
Dev Set	0.9607	0.9117	0.9356
Test set	0.9573	0.9133	0.9348

Table 6: Neural Model Scores

8.1 Data Processing

In this phase of data preprocessing, we used dictionaries for train, dev and test sets which contain the sentences in the form of list of word tuples as mentioned in the section of Data Preparation. We also created vocabularies for words, tags and characters present in our dataset. Also, we prepared a numpy array which contains the embeddings for tokens using fastText Spanish embeddings.

8.2 Comparison with Baseline Model

Below are the results for all the implemented models -

	Dev Set			**Test Set**		
	P	**R**	**F1**	**P**	**R**	**F1**
Simple Baseline	0.6292	0.4254	0.5076	0.628	0.4238	0.5061
Baseline	0.8868	0.8196	0.8519	0.8915	0.8335	0.8615
Neural Model	0.9607	0.9117	0.9356	0.9573	0.9133	0.9348

8.3 Confusion Plot

Confusion plot after running the model for 5 epochs -

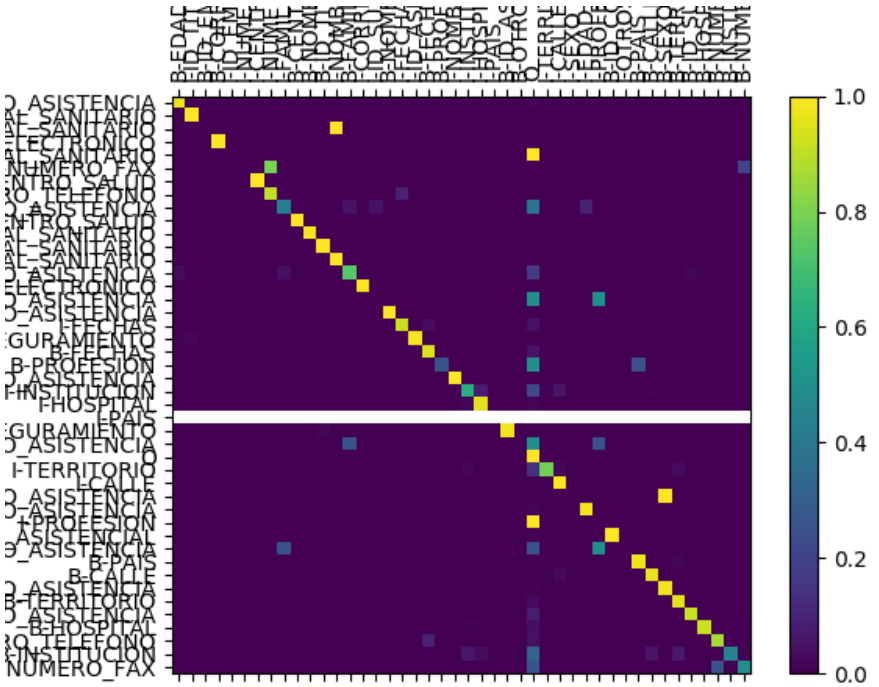

Figure 4: Confusion Plot

9 Error Analysis

The neural model is not able to perform well for predicting the following types of things-

1. Skewed distribution for category 'ID_SUJETO_ASISTENCIA':

As evident from confusion plot, we can see that the model is performing poorly on the category 'ID_SUJETO_ASISTENCIA'. On analysis, we found out this category has a skewed distribution in the training set. In gold files, the category 'ID_SUJETO_ASISTENCIA' has been assigned to numbers in majority of the documents and only some times to text. Therefore, our model learnt to assign this category to numbers only.

A snippet of gold files containing the category 'ID_SUJETO_ASISTENCIA':

```
S0211-57352011000100008-1.ann:T21   ID_SUJETO_ASISTENCIA 53 61  78465239
S0211-57352013000300012-1.ann:T17   ID_SUJETO_ASISTENCIA 646 670    la mayor de
dos hermanas
S0211-57352013000300012-1.ann:T31   ID_SUJETO_ASISTENCIA 69 76  4396589
S0211-69952009000500014-1.ann:T19   ID_SUJETO_ASISTENCIA 51 58  7846523
S0365-66912006000600011-1.ann:T15   ID_SUJETO_ASISTENCIA 2015 2023  el menor
S0365-66912006000600011-1.ann:T32   ID_SUJETO_ASISTENCIA 53 60  1456764
S0365-66912006000700009-1.ann:T24   ID_SUJETO_ASISTENCIA 64 71  4768426
S0365-66912006000700011-1.ann:T22   ID_SUJETO_ASISTENCIA 71 78  8347523
```

A snippet of predicted files containing the category 'ID_SUJETO_ASISTENCIA':

```
S0211-57352011000100008-1.ann:T3    ID_SUJETO_ASISTENCIA 53 61  78465239
S0211-57352013000300012-1.ann:T3    ID_SUJETO_ASISTENCIA 69 76  4396589
S0211-69952009000500014-1.ann:T3    ID_SUJETO_ASISTENCIA 51 58  7846523
S0211-69952009000600023-1.ann:T3    ID_SUJETO_ASISTENCIA 49 57  96147325
S0211-69952011000400013-1.ann:T3    ID_SUJETO_ASISTENCIA 69 76  7834564
S0211-69952011000500011-1.ann:T3    ID_SUJETO_ASISTENCIA 50 59  784123665
S0211-69952011000500013-1.ann:T3    ID_SUJETO_ASISTENCIA 49 58  126348975
S0211-69952012000700031-1.ann:T3    ID_SUJETO_ASISTENCIA 54 61  1526987
```

2. No heterogeneity in the data for category 'HOSPITAL':

On analysing the errors related to the category 'HOSPITAL', we found out that the label 'HOSPITAL' is assigned to those terms in the training data which contain the term 'HOSPITAL' in it. This means that we don't have variety of examples for this category in our training set. Thus, the model learnt to predict the category 'HOSPITAL' only if the token itself contains this term.

A snippet of gold files containing the category 'HOSPITAL':

```
S0211-69952009000500014-1.ann:T3 HOSPITAL 2658 2689 Hospital General de Ciudad
Real
S0211-69952011000400013-1.ann:T5 HOSPITAL 4768 4794 IIS-Fundación Jiménez Díaz
S0211-69952011000500011-1.ann:T3 HOSPITAL 1987 2024 Hospital Universitario de
Guadalajara
S0211-69952011000500013-1.ann:T5 HOSPITAL 3579 3594 Virgen Macarena
S0211-69952012000700031-1.ann:T5 HOSPITAL 2271 2313 Hospital Universitario Central
de Asturias
S0211-69952013000200019-2.ann:T4 HOSPITAL 1483 1502 Hospital de Manises
S0211-69952013000500019-1.ann:T6 HOSPITAL 4414 4436 Hospital 12 de Octubre
```

A snippet of predicted files containing the category 'HOSPITAL':

```
S0211-69952009000500014-1.ann:T19 HOSPITAL 2658 2689 Hospital General de Ciudad
Real
S0211-69952011000500011-1.ann:T16 HOSPITAL 1987 2024 Hospital Universitario de
Guadalajara
S0211-69952012000700031-1.ann:T17 HOSPITAL 2271 2313 Hospital Universitario Central
de Asturias
S0211-69952013000200019-2.ann:T20 HOSPITAL 1483 1502 Hospital de Manises
S0211-69952014000400019-1.ann:T22 HOSPITAL 2666 2691 Hospital de la Santa Creu
S0211-69952015000500015-2.ann:T23 HOSPITAL 1600 1620 Hospital de Poniente
S0212-16112006000100018-1.ann:T17 HOSPITAL 2302 2349 Hospital General Universitario
Gregorio Marañón
```

S0211-69952011000500013-1.ann:T5 O 3579 3594 Virgen Macarena

10 Conclusion

In the project, we tried different approaches for the task of de-identification of PHI in Spanish clinical records. We started with simple rule-based model and then moved on to LinearSVC. We then tried a system that is a combination of rule-based, LinearSVC and static dictionaries of Spanish names and locations. Later, we trained a neural network that uses Bi-LSTM and CRF for named entity recognition. The neural model performed best for us on the given dataset.

11 Future work

As a next step, we are planning to build a system that is a combination of rule based model and our best performing neural model. For the structured text, we will use the prediction made by our rule based system and for the unstructured text, we will go with the neural model predictions. We intend to run few more experiments and submit our best system to the ongoing MEDDOCAN contest by May 17, 2019.

12 Acknowledgement

We would like to thank Prof. Chris Callison-Burch, and João Sedoc for their guidance throughout the project.

13 Code Link

Please refer to 'finalSubmission' folder in the following GitHub repository.
GitHub link:
https://github.com/ionur/Identifying-Words-to-Anonymize-MEDDOCAN

14 Presentation Link

Presentation link:
https://docs.google.com/presentation/d/1fWq5xAv3SrCDAZkWzhFfLejaNQbNutaWzkpx2fmtTmY/edit#slide=id.p

References

[1] Spanish location names wikipedia. https://en.wikipedia.org/wiki/List_of_municipalities_of_Spain, 2019.

[2] Spanish male and female names. https://github.com/substack/provinces, 2019.

[3] Thomas Brox Røst Arild Faxvaag Øystein Nytrø Torbjørn Nordgård Martin Thorsen Ranang Amund Tveit, Ole Edsberg and Anders Grimsmo. Anonymization of general practioner medical records. HelsIT, 5 pages, 2019.

[4] Active Competition. Classifying medical health records using svm. `http://temu.bsc.es/meddocan/`, 2019.

[5] Active Competition. Link to the dataset. `http://temu.bsc.es/meddocan/index.php/data/`, 2019.

[6] Róbert Busa-fekete György Szarvas, Richárd Farkas. State-of-the-art anonymization of medical records using an iterative machine learning framework. Journal of the American Medical Informatics Association Volume 14 Number 5 Sept / Oct 2007, 7 pages, 2019.

[7] Rema Padman Kaung Khin, Philipp Burckhardt. A deep learning architecture for de-identification of patient notes: Implementation and evaluation. arXiv:1810.01570 [cs.CL], 3 Oct 2018, 15 pages, 2019.

[8] Xiao-Bai Li, "Anonymizing Jialun Qin, and Sharing Medical Text Records. Anonymizing and sharing medical text records. Inf Syst Res. Author manuscript; available in PMC 2018 March 19, 47 pages, 2019.

[9] Eduard Hovy Xuezhe Ma. End-to-end sequence labeling via bi-directional lstm-cnns-crf. `https://arxiv.org/pdf/1603.01354.pdf`, 2016.

[10] Ian Roberts-George Demetriou Mark Hepple Yikun Guo, Robert Gaizauskas. Identifying personal health information using support vector machines. 2006, 5 pages, 2019.