Universitat Politecnica de Catalunya
Facultat d'Informatica de Barcelona
Computer Science Department

Bootstrapping for electronic health record anonymization with minimal supervision

Salvador Medina Herrera

Supervisor: Jordi Turmo Borràs

Master Thesis Report for the Master of Artificial Intelligence of
the Facultat d'Informatica de Barcelona, January 2017

Abstract

Electronic health records are an important source for the research and study of diseases, treatments and symptoms. However, due to data protection laws, information that could potentially compromise privacy must be excluded before making use of them. The precise identification of these pieces of information is then mandatory. Supervised learning has often been used for creating anonymization systems, but the cost of building the required corpora can be prohibitive. In this work, we propose a bootstrapping strategy so as to enrich anonymization models for Catalan health records taking profit of huge sets of unlabeled documents. We demonstrate how models that use word-embeddings as input features greatly benefit from applying this strategy even when starting from small or biased training corpora.

Acknowledgements

I would like to express my sincere gratitude to:

- Jordi Turmo Borras, director of this Master thesis
- Horacio Rodríguez Hontoria, coordinator of the Advanced Natural Language Processing subject
- My family and friends

‘The problem with losing anonymity is that you can never go back.’

Marla Maples

Contents

Abstract **i**

Acknowledgements **iii**

1 Introduction **1**

1.1 Motivation 1

1.2 State of the Art 2

1.3 Contributions 5

2 Definition of our anonymization task **6**

2.1 Introduction 6

2.2 De-identification and Anonymization 6

2.2.1 k-anonymity 7

2.2.2 De-identification methods 7

2.3 Anonymization of electronic health records 8

2.3.1 Directives for privacy protection in health records 8

2.3.2 Anonymization as sequence labeling 10

2.4 IDIAP's corpus of electronic health records 11

3 Methodology **12**

3.1 Introduction 12

3.2 Semi-supervised learning 12

3.2.1 Self-Learning 13

3.3 Vectorial Word Representations 15

3.3.1 Word-Embedding Models 16

3.3.2 Word-Embedding Clusters 17

3.4 The Conditional Random Field Model 17

3.4.1 Conditional Random Fields 18

3.4.2 Implementation 19

3.5 The Bilinear Long Short-Term Memory Model 21

3.5.1 Long Short-Time Memory networks 21

3.5.2 Implementation 23

3.6 Observation-based learning 25

3.6.1 Binary Dictionary Model 26

3.6.2 Frequency Divergence Model 27

3.6.3 Dictionaries 27

4 Experiments and results **29**

4.1 Experimental setup 29

4.2 Results using a small training corpus 30

4.3 Results with a biased learning corpus 30

4.4 Comparison with active-learning 32

4.5 Results with no manually labeled training corpus 34

5 Conclusion **36**

5.1 Summary of Thesis Achievements 36

5.2 Future Work 37

A Personal Health Information categories according to the Information Portability and Accountability Act **39**

B The corpus of Catalan health records **41**

C Examples of PERSON and LOCATION in the IDIAP dataset **43**

Bibliography **43**

List of Tables

1.1 Examples of the free-text section of the *i2b2 2006*, *2016 CEGS N-GRID*, *MIMIC-II* and *IDIAP* corpora. The first 3 sets also include an structured *xml* header including the patient's and doctor's names, along with other medical information. Target named entities have been anonymized using generic names but maintaining their capitalization and number of tokens. 3

2.1 Statistics of PHI in the validation corpus. 11

3.1 Token normalization strategies used for building the word-embedding models and count of unique tokens. Median cluster size for 128, 1024 and 8192 clusters. 16

3.2 Precision and recall of predictions made by the dictionary-based models with and without a confidence threshold. Using strict evaluation. 28

4.1 Statistics of the datasets used for training and validation 29

4.2 Baseline F_1 score for PERSON and LOCATION obtained using 7-fold cross validation of the validation corpus. CRF (*wc*) means that word clusters were used with the CRF model. 30

4.3 F_1 score for PERSONand LOCATIONobtained using a small training corpus. *wc*, lm, mph, pos and c stand for word clusters, lemmas, morphology, POS tags and capitalization respectively. fhu, ls and fd stand for few hidden units, long sequences and few selected documents respectively. Trend indicates whether the combined F_1 tended to decrease or increase using our semi-supervised strategy. . 31

4.4 F_1 score for PERSON and LOCATION obtained using the DIP-2.0 (biased) training corpus. *wc*, lm, mph, pos and c stand for word clusters, lemmas, morphology, POS tags and capitalization respectively. fhu, ls and fd stand for few hidden units, long sequences and few selected documents respectively. Trend indicates whether the combined F_1 tended to decrease or increase using our semi-supervised strategy. 32

4.5 F_1 score for PERSON using handcrafted observation-based model as a base. . . . 35

B.1 Statistics of the corpus of Catalan health records of 2013 42

C.1 Instances of categories PERSON and LOCATION in the IDIAP dataset. All instances have been previously shuffled. 44

List of Figures

2.1 Example of BIO encoding a sentence 11

3.1 Example of a linear-chain CRF applied to sentence tagging. 18

3.2 Schematic representation of a LSTM cell. Source: LSTM cell schematic (Graves, 2013) - ResearchGate. 22

3.3 Overlapping and padding strategies to keep input sequences' length constant. 25% of the beginning and end of the sub-sequences is overlapped, output is combined so that context before and after are maximized. 24

3.4 Layout of the BiLSTM-CRF network used in this project. LSTM blocks are composed by $\frac{N}{2}$ memory units fully connected to all elements of the feature vector. A dropout factor or 0.5 is applied to both LSTM blocks. 26

4.1 Evolution of precision, recall and F_1 score for two different executions of the self-learning algorithm trained with the DIP-2.0 corpus. Top: CRF model. Bottom: BiLSTM-CRF model. 33

4.2 Combined recall, precision and F_1 score in the validation and test corpus achieved for each active-learning iteration. The number of retrieved documents at each iteration was set to 250 and ranking was inversely proportional to confidence. . . 34

Chapter 1

Introduction

1.1 Motivation

The analysis of clinical reports is crucial for the research of human life sciences. Studying these reports facilitates the tasks of doctors and researchers, as they represent a reliable source for determining the large-scale effects of both diseases and treatments. It is hence important that hospitals make them publicly available to both personnel and scientific researchers.

However, this kind of documents often include personal information about patients and medical staff, which would break data protection laws if made publicly available. Spanish data protection laws explicitly include these documents in the group of specially protected data[1]. As a result, the publication of these reports is conditioned to a previous step of anonymization. Electronic health records should then be freed from first and last names of patients, medical staff and doctors; identification or sanitary card numbers and social security codes; telephones and e-mails; and both public and private clinics and centers, addresses and geographic locations.

This process is sometimes made manually by specialized curators, whose cost could be prohibitive for most medical institutions. Alternatively, document anonymization models based on natural language processing and named entity recognition can be applied for automatizing

[1]Datos Especialmente Protegidos - Agencia Española de Protección de Datos

this task. Nevertheless, due to health records not being written in standard language, using general-purpose NERC tools is not viable and context-specific models should be learned. This can also be quite challenging though, since large manually labeled corpora is not widely available, specially for languages other than English.

1.2 State of the Art

Over the past years, many anonymization and de-identification approaches have been proposed in the literature, mostly focused on the guidelines from the U.S. Health Insurance Portability and Accountability Act (HIPAA) for protected health information (PHI). This work can be roughly categorized in those approaches that combine structured data and free text, and those strictly applied to natural language unstructured data.

Due to the sensitive nature of health reports, publicly available datasets are limited and most of them correspond to shared challenges such as Informatics for Integrating Biology to Bedside (*i2b2*) de-identification challenges in 2006 [25] and 2014 [22] and the 2016 CEGS N-GRID shared tasks[21], or are re-purposed datasets such as nurses' notes extracted from the Multiparameter Intelligent Monitoring for Intensive Care (MIMIC-II) dataset [13]. Other work make use of private datasets, such as Beth Israel Deaconess Medical Center [26]. Samples from the aforementioned challenges are shown in table 1.1.

Most recent work tackles the anonymization problem by applying common Named Entity Recognition (NER) methods, specially Conditional Random Fields (CRF) to identify sequences of tokens that correspond to each one of the PHI categories. Yang et Al. [28] scored first in the *i2b2 2014* challenge by using CRFs with lemmas, part-of-speech and morphological features in a window of 3 tokens; and combine them with context specific gazetteers to identify PHI. The model by Yang also included several handcrafted post-processing rules to ensure coherency between entities and identify entities following formal patterns such as dates and telephone numbers. Dehghan et Al. [4] scored second in the same challenge using a very similar model also based on a CRF tagger, a set of manual rules and several gazetteers; resolving

i2b2 2006	Works in programming at **NewCo**. Formerly at **Acme**. He has remote travel hx to the **High Street**, more recent global History of Present Illness: Pt is a 59 yo **Homer** male, with who was admitted to **San John Doe Hospital** following a syncopal nauseas and was brought to **John Doe ED**. Five weeks ago prior Anemia: On admission to **John Hospital**, Hb/Hct: 11.6/35.5.
2016 CEGS	[...] Developmental History/ Family of Origin Developmental History: Grew up in **Springfield**, CO. Parents divorced when she was 3. After college, movde in with grandparents to be a care giver. Lived there until 2 years ago. Currently lives in **Shelbyville** with her husband and two step-children (**Marge**, aged 2; **Lisa**, aged 5)Past verbal, emotional, physical, sexual abuse: No Social History Marital Status: Married Does patient have any children: Yes 2 children (ages 2 and 5)Interpersonal Interactions/ Concerns: -grief after death of grandparents. -struggling with prioritizing amily and self-interestsGambling behavior: No [...]
MIMIC-II	**Homer** admitted in transfer from **John Hospital** on XXXX-XX-XX following a clinic visit where he c/o sob, ruq pain, and possible r/o chf. pt admitted to ccu on XX-XX for increased aggitation, twitching, periods of unresponsiveness, arf illustrated by a creatnine of 4 (1.2 1 month ago), and elevated liver enzymes. pt of dr. **Julius Hibbert**.
IDIAP	XXXX|XXXXXXXX|X|Ginec / results ..Seguimiento en Centro privado **Acme Corporation**..Nuligesta .Con antecedentes de trastornos menstruales ..ECO anterior informa presencia de polipo,proceder que repiten post menstruación..Aporta eco tv evolutiva realizada en centro privado **Clínica Acme Corporation** ;Dr **JULIUS Hibbert** diagnóstico de Polipo endometrial..Plan Derivación a nivel III/ exéresis de polipo mediante histeroscopia y de acuerdo a protocolo.

Table 1.1: Examples of the free-text section of the *i2b2 2006*, *2016 CEGS N-GRID*, *MIMIC-II* and *IDIAP* corpora. The first 3 sets also include an structured *xml* header including the patient's and doctor's names, along with other medical information. Target named entities have been anonymized using generic names but maintaining their capitalization and number of tokens.

the overlapping annotations by priority.

Dernoncourt et Al. [5] could outperform the model by Yang et Al. by using Bidirectional Long Short-Term Memory (BiLSTM) networks, optionally in combination with CRF. It used character embeddings in combination with both context-specific and general-purpose word embeddings as input features. Dernoncourt concludes that adding a final CRF layer to the BiLSTM model boosts the network's performance in terms of F_1 score when the number of training documents is small, but slightly penalizes it for large datasets.

In addition to LSTM, other Recurrent Neural Network (RNN) models have been applied to the task. Shweta et Al. [20] propose two taggers using the RNN models described by Elman (E-RNN), with recursion within the hidden layer; and Jordan (J-RNN), with the context information flowing from the output layer to the hidden layer. In addition to word-embedding features within a window of 3 tokens, Shweta also uses *n-grams* and *POS* tags. As in the model by Dernoncourt, the final sequence tagging is also performed by a CRF. Despite the more extensive list of features, the BiLSTM-CRF model outperforms both E-RNN and J-RNN in the *i2b2 2014* dataset.

As the majority of participants in the *2016 CEGS N-GRID shared task 1*, Hee et Al. [11] use CRF in combination with post-processing rules. However, the system proposed by Hee could score second place by combining a rule-based classifier with two CRF taggers into a hybrid system. One of the two CRF taggers is applied at token level and the other at character level. Rule-based post-processing is also evaluated to correct common errors, and finally agreement is enforced between entities in the same document with the same string value.

Previous work have also explored the use of Support Vector Machines (Guo et Al. [7]), and boosting with decision trees (Szarvas et Al. [23]). In both cases, the structured heading section of the documents in the *2006 i2b2* dataset is used as additional features in conjunction with context-specific gazetteers. Each token is assigned a Begin, In, Out (BIO) tag, which is common in sequence-based Named Entity Recognition models.

1.3 Contributions

This work tries to improve the results obtained by the models based on CRF and Bi-LSTM, taking profit of the large amount of unlabeled data available. We apply statistical semi-supervised techniques based on the confidence of the automatically labeled sequences to find reliable examples with no manual supervision. This approach has proven to be successful in other natural language processing tasks such as sentiment analysis of tweets [27].

Moreover, we study how it is influenced by the initial training corpus. In particular, we see how results are affected by a small and biased training corpora and how it can be overcame with semi-supervision. Finally, we explore two gazetteer-based alternatives to build an initial training set with no manual interaction.

Another remarkable contribution is the fact that we use a completely unstructured corpus of health records written in a mix of Catalan and Spanish. As opposed to most work in the literature, which focuses on English documents that contain a reliable and well-structured header section. Likewise, these records are mostly taken from primary care services and combine admission, progress, operative and discharge notes.

To sum up, this work tries to moderate one of the main limitations of state-of-the-art models, specially for records in languages other than English, which is the need for a relatively large and hence expensive labeled corpus. We also study whether or not semi-supervision can be applied to a simple observation-based model and no manually labeled training set.

Chapter 2

Definition of our anonymization task

2.1 Introduction

This chapter introduces the anonymization task. Section 2.2 explains what de-identification and anonymization are and describes some widely-used strategies. Section 2.3 presents the problem of anonymizing electronic health records and the labeling criteria that we are adopting, with the justification of how anonymization can be reduced to a sequence labeling task. Section 2.4 briefly describes the characteristics of IDIAP's corpus of electronic health records.

2.2 De-identification and Anonymization

When considering the problem of how to protect personal information in written documents, there are multiple alternatives that could potentially prevent someone from being able to identify who was originally referenced in the documents.

De-identification of data refers to the process of removing or hiding any personal information in a way that minimizes the risk of unintended exposure of the identity of whoever is referenced in the document. Anonymization is a particular and more restrictive case of de-identification that produces data where personal records cannot be linked back to their original target, as

the required variables to do so are not included. De-identified records on the other hand may include preserving identifying information which can only be re-linked by a trusted party in certain situations. De-identification assumes that it might not be possible to remove all linkable elements and hence all risk, but it is considered successful when there is not reasonable way to believe that the remaining information in the texts can be useful to identify any originally referenced individual.

2.2.1 k-anonymity

A commonly used de-identification criterion is k-anonymity. This criterion stipulates that each record in a dataset is similar to at least another $k-1$ records on the potentially identifying variables. For example, if $k = 5$ and the potentially identifying variables are age and gender, then a k-anonymized dataset has at least 5 records for each value combination of age and gender. For small k, these variables could then be understood as Quasi-Identifiers (QS), because they could be potentially used to identify the individual they refer to, even though more specific variables such as the name and social security number are hidden.

Ensuring k-anonymity for large k is very important in de-identification tasks, since anonymization systems have to make sure that all the information that is left unchanged in the document cannot be used as quasi-identifiers. This can represent a trade-off, as some of this quasi-identifying information such as age might be needed for research purposes.

2.2.2 De-identification methods

Depending on the type of data that has to be hidden or the ambit and risk of the original document, there are multiple methods for de-identifying documents. In the particular field of electronic health records, Nelson [14] describes several strategies applicable to multiple kinds of information that could potentially compromise privacy at different levels, and discusses about their implications. Some of the most relevant and widely-used ones are summarized below.

- **Suppression**: Completely remove a piece of information. This can be used for example for very rare diseases that could be easily linked to an individual.
- **Randomization**: Replace the entities by a randomly generated value. For example, replace the name of a patient by another one extracted from a dictionary.
- **Shuffling**: Swap instances of entities of the same type among the set of documents.
- **Surrogate**: Replace entities by an identifier which is constant among all appearances of the same entity but cannot be associated to the original value. For example, replacing a variable by its hash.
- **Aggregation**: Aggregate rare identifiers into bigger groups. For example, replace small village's names by their respective province.
- **Character masking or scrambling**: Replace some characters of an entity by others or rearrange them. For example *JOHN* could be changed to *HONJ* or *J****.
- **Blurring**: Convert continuous variables to categorical elements. For instance, an age of 16 could be changed to *teenager*.

2.3 Anonymization of electronic health records

In recent years there has been an active research in the field of life sciences, that often requires the use of real health records. Nonetheless, getting access to this data can be complicated, as in most legislations it is conditioned to a previous anonymization step. This is why research in the field of anonymization of health notes has also seen a significant leap; and public organisms have started issuing challenges and anonymization guidelines.

2.3.1 Directives for privacy protection in health records

Each country has its own data protection laws and anonymization systems have to be adapted to each legislation. Most of the work in the field has been focused on American health records

and hence fit to the guidelines of the Health Information Portability and Accountability Act (HIPAA), which are listed in appendix A.

In Spain, data protection is ensured by the *Ley Orgánica de Protección de Datos* (LOPD). The LOPD states that documents containing personal information cannot be distributed unless there is an explicit consent from the people referenced in them or all information that could be potentially used to identify these people has been completely removed. Elustondo et al. [6] describes additional technical and ethical assumptions included in the LOPD and reasons about their implications for medical research.

Anonymization guidelines

In our work, we follow the anonymization directives given by the *Institut Universitari d'Investigació en Atenció Primària* (IDIAP), a medical research center subordinated to the *Institut Català de la Salut* (ICS). We are required to replace the target entities' phrases by their respective category name. The categories that must be replaced, which are a subset of those defined by the HIPAA, are listed below:

1. PERSON: Name or surname of a patient, relative, medical staff or any other person mentioned in the report.
2. LOCATION: Physical locations or geographic subdivisions including street address, city, county, precinct, ZIP code, etcetera. This also includes public locations such as hospitals, clinics, schools and others.
3. TELEPHONE: Digits of a phone number
4. EMAIL: E-mail address
5. DNI: Spanish *Documento Nacional de Identificación*
6. SOCIAL_SECURITY_ID: Spanish social security number
7. SANITARY_CARD_ID: Catalan sanitary card number

From all the aforementioned categories we only consider PERSON and LOCATION, since all the others have a formal structure and preliminary tests showed that they can be successfully identified by using a set of regular expressions. This was also the case in other anonymization models presented in section 1.2. Several examples of the aforementioned categories in the dataset are listed in appendix C.

2.3.2 Anonymization as sequence labeling

Anonymization, and more generally de-identification, are closely related to named entity recognition, since most of the target categories to be anonymized are some kind of named entity. Hence, most techniques applied to named entity recognition can also be applied to anonymization as an initial step to remove personal information from texts.

Once the bounds of each noun phrase that contains personal data are identified, there are multiple options to hide this information. Depending on the requirements, one could replace all instances by a placeholder (*John Doe* $\Rightarrow$ `PERSON`); or a random synthetically generated replacement (*John Doe* $\Rightarrow$ *Jane Roe*). But the main challenge is to be able to identify such phrases.

The task of identifying phrases can be viewed as a sequence labeling problem in which phrases are denoted by assigning labels to individual words indicating whether or not the word is part of a phrase of a particular named entity type.

A common encoding is to use two labels for each type of phrase: one indicating that the word begins a phrase and another indicating that the word is within or ending a phrase. This is known as BIO encoding. Figure 2.1 provides an example of the BIO tags assigned to a short excerpt from a medical report.

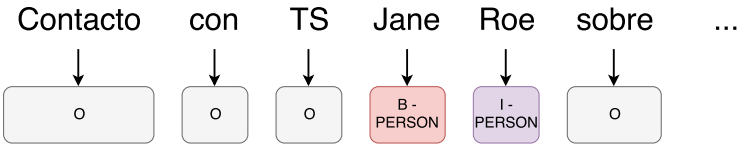


Figure 2.1: Example of BIO encoding a sentence

NE	Support	Frequency	Avg. Tokens
LOCATION	678	0.0112	1.844
PERSON	257	0.0028	1.210
TELEPHONE	6	0.0001	2.5

Table 2.1: Statistics of PHI in the validation corpus.

2.4 IDIAP's corpus of electronic health records

The IDIAP corpus of Catalan health records of 2013 is a compilation of short health notes mostly taken from primary care services. The entries do not contain any structured header, and combine admission, progress, operative and discharge notes, covering multiple medical fields. Appendices B and C give more details about the IDIAP corpus and contain some samples of health records.

Table 2.1 shows the frequency of some of the PHI in the validation corpus. As we see, the frequency of some relevant categories is very low, meaning that the cost of building corpora for training and evaluating a supervised model is very high. This justifies the need for alternative semi-supervised learning algorithms that can benefit from unlabeled corpora as the one we are presenting.

Chapter 3

Methodology

3.1 Introduction

This chapter describes the semi-supervised methodology used as well as the supervised learning models it has been applied to. Section 3.2.1 introduces the self-learning approach that we propose and gives an overview of how documents are divided and bootstrapped. The CRF and BiLSTM-CRF models and the features that have been considered for each one of them are described in Sections 3.4 and 3.5. Section 3.6 we show the two basic observation-based taggers used when no manually labeled training corpus is provided.

3.2 Semi-supervised learning

This project revolves around the idea of using a previously learned anonymization model to identify relevant documents from the large set of unlabeled data. In doing so, we can find new patterns that so as to enrich the training corpus to learn an improved anonymization model. The selection is done based on the confidence of the classifier. This is a classical semi-supervised learning approach known as self-training or self-labeling [24].

While self-training approaches do not perform as well as fully supervised models, they are a

good alternative when the cost of manually labeling a big-enough training corpus is prohibitive. The final performance is conditioned to the quality and generality of the initial classifier: those with high precision but low recall are usually not fruitful as the patterns obtained may already be found in the training set; and those with high recall but low precision might increase the error rate of the bootstrapped training set.

Co-training is a self-training alternative that joins the output of two or more uncorrelated classifiers (views of the data) to refine the confidence of the candidate documents. Views often use a completely different feature set. However, it can suffer from the same issues as classic self-training [19]. It also requires more work, since two different uncorrelated models have to be defined and trained. This is pretty straightforward when we have two different sources such as the structured header and the free text of a health record, but harder for the type of documents that we are dealing with.

3.2.1 Self-Learning

We apply self-learning to CRF and BiLSTM-CRF models in batches of 50000 documents, extracted from the the health notes issued during the month of November. Model $U(t-1)$ is applied to batch $B(t)$ and then documents with the highest confidence are selected. The maximum amount of selected documents and the minimum confidence value are defined as parameters. The proportions of examples of the different classes can optionally be enforced.

A new model $U(t)$ is learned by combining all selected examples of previous batches with the manually labeled training corpus. This process is then repeated until the model converges according to the selected convergence criteria or until the maximum number of iterations is reached. A pseudo-code of the algorithm is described in algorithm 1.

Function *fit_model* trains a new tagger at each iteration of the execution. It receives the iteration index i as a parameter so that the model can be changed for different iterations of the same run, which can be interesting to prevent stagnation. The ability of changing the model is used when an observation-based model is applied in iteration 0, which doesn't make use of a

Algorithm 1 Pseudo-code of the implemented self-learning algorithm. C_{min} is the confidence threshold, It_{max} is the maximum iteration and B_i represents the *i-th* unlabeled batch.

$i \leftarrow 0$
$m \leftarrow fit\_model(X_{train}, Y_{train}, 0)$
$evaluations \leftarrow \emptyset$
while $i < It_{max} \wedge \neg converges(evaluations)$ **do**
  $Y_{b_i} \leftarrow run(m, B_i)$
  $X_{selected}, Y_{selected} \leftarrow select\_examples(B_i, Y_{b_i}, k, C_{min})$
  $X_{train} \leftarrow X_{train} \cup X_{selected}$
  $Y_{train} \leftarrow Y_{train} \cup Y_{selected}$
  $m \leftarrow fit\_model(X_{train}, Y_{train}, i)$
  $evaluations \leftarrow evaluations \cup evaluate(m, X_{validation}, Y_{validation})$
  $i \leftarrow i + 1$
end while
return m

training corpus; and is replaced in the successive iterations by a supervised learning model.

Function *evaluate* runs the model m in validation set and computes the F_1 score. Function *converges* determines when to stop iterating based on the achieved score.

In our experiments, we set the maximum number of iterations to $It_{max} = 20$. In order to save time, we define the *converges* function so that convergence is reached if the F_1 score in the validation corpus has not improved in the last 3 iterations (stagnation). The constant k limits the maximum amount of selected documents and is set to $k \in [2000, 5000, 10000, 20000]$ in our experiments. A different confidence value C_{min} can be defined for each classification model. We use $C_{min} = 0.85$ for the CRF and BiLSTM-CRF models and $C_{min} = 0.5$ for the frequency-divergence model.

Selection of new examples

Function *select_examples* receives a list of tagged documents with the confidence value given by the previous model and outputs a subset of them. Sections 3.4.2 and 3.5.2 describe how this confidence value is computed for each model.

No documents with confidence lower than the threshold are chosen and the output list is trimmed to a maximum length k. It also ensures that the proportions of selected examples of

each class fits the requirements so that population is maintained in the resulting training set.

Supervised models

Our self-learning algorithm uses unlabeled data and heuristics to add new examples to the training set, with the objective of finding new relevant features previously overlooked to improve the original model. Hence the choice of the supervised learner that is trained with the new training set is very important.

At each iteration of the *Self-Learning* algorithm, a new model is learned from the automatically enlarged training set. Any supervised learning model could be used at this point, but we focused on the two models that have shown better performance in the most recent anonymization challenges: Conditional Random Fields and Bilinear Long Short-Time Memory artificial neural networks, as they achieve state-of-the-art performance for PHI anonymization tasks. For both of the models, we applied the 3 word-embedding models learned from the unlabeled dataset described in 3.3.

3.3 Vectorial Word Representations

One of the input features that we used for both of our supervised learning models are vector representations of individual words. Word embeddings add a semantic component to the tokens that have proven to lead to better accuracy in named entity recognition tasks. What is more, neural network models such as LSTM require the input to be vectorized and due to their being a more compact representation, word-embeddings are often preferred over traditional 1-hot encoding of tokens.

Learning independent representations for word types from a limited NER training data is a hard task. Consequently, generic word representations pre-trained from a large content-independent dataset are often used. One such example is *fastText* from Facebook Research[2], which is built from entries in Wikipedia for 294 languages.

Normalization	# Tokens	Median cluster size		
		$K = 128$	$K = 1024$	$K = 8192$
None	582047	3226.0	384.0	42.0
Lowercased	441679	2643.5	296.0	30.0
Lowercased & no accent marks	413238	2450.5	274.0	28.0

Table 3.1: Token normalization strategies used for building the word-embedding models and count of unique tokens. Median cluster size for 128, 1024 and 8192 clusters.

In our case, however, we have a relatively big untrained corpus of 631 million words, which seems like a reasonable amount to build a context-specific word-embedding model. Our intuition in doing so is the fact that relevant words such as names and locations, which may be very varied individually, appear in regular contexts in the large corpora.

3.3.1 Word-Embedding Models

Some of the input features of the supervised models that consider in our work are defined as a vectorial representation of words (word-embeddings). Given the particular terminology and characteristics of the documents in the corpus, instead of using a pre-trained general-purpose model, we build three different word-embedding models from the full unlabeled corpus. These models are learned using *Python's* word2vec library, an implementation of Mikolov's *word2vec* [12].

Preliminary tests were carried out for different vector and window sizes ($N \in [50, 100, 300]$ and $w \in [3, 5, 10]$). Given the preliminary results obtained, out of the scope of this project, we decided to set the length of the feature vectors to $N = 100$ and the window size and count threshold for words to 5.

In order to generate the models, all documents are tokenized using *FreeLing's*[1] Spanish tokenizer [16]. The three models differ in the token normalization strategy used. A summary of the statistics of the three resulting models is shown in table 3.1.

[1]FreeLing, an open source language analysis tool suite developed at the Universitat Politècnica de Catalunya

3.3.2 Word-Embedding Clusters

Having chosen a word embedding model, another important design choice regards the method by which the features are incorporated into the classification model. Directly applying the feature vector as an input may not be a good choice for some language modeling tasks, as it was shown by Bansal et al. [1]. This is specially true for simpler models such as Naïve Bayes, which considers all input features to be mutually independent.

One solution is then to cluster the token's feature vectors in advance and use a 1-hot encoding of the cluster index as the input feature. Note that the number of clusters and the distance between them can affect the classifier's accuracy: we may loose relevant information if we use too few clusters but we might not have enough training instances to cover all relevant clusters when there are too many of them.

In this project, we use *K-Means* clustering with $K \in [128, 1024, 8192]$ to build such clusters. Table 3.1 shows the median of clusters' sizes for the different token normalization strategies and number of clusters K. Note that even for $K = 8192$, no cluster is empty and the median of the cluster sizes is similar to its mean (from 20% to 40% smaller); which means that the number of clusters is not excessive and words are well distributed among them.

3.4 The Conditional Random Field Model

Most of the state of the art work on named entity recognition and anonymization is based on CRFs using both morphological and gazetteer based features. These kind of models are easy to understand, as they compute the probability of a labeled sequence of tokens conditioned to the considered features. Hence, given a certain feature, it is easy to determine how relevant it is for the tagger. Work in the subject has also proven that using word embedding features - discretized or clustered - can improve its performance for named entity recognition tasks.

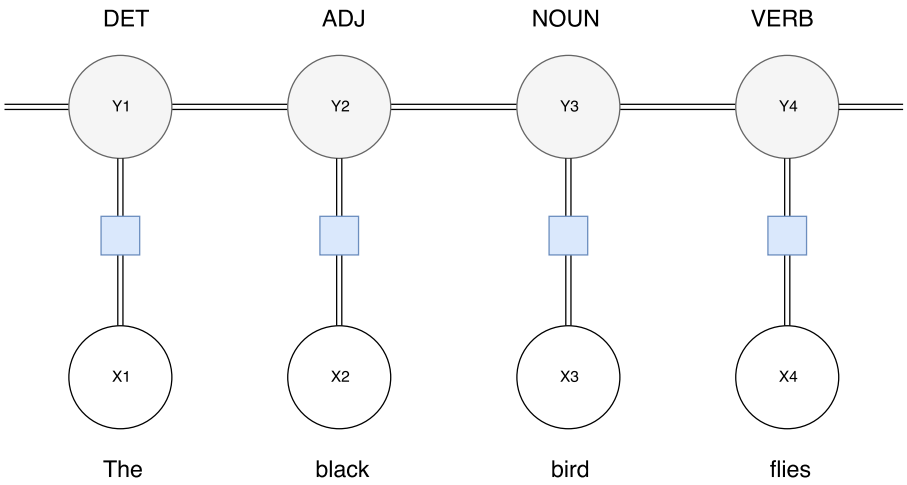


Figure 3.1: Example of a linear-chain CRF applied to sentence tagging.

3.4.1 Conditional Random Fields

Conditional Random Fields are probabilistic models used for labeling sequential data, first proposed by Lafferty et al. in 2001 [10]. Two of the most widely used applications of CRFs are Part Of Speech (POS) tagging and Named Entity Recognition (NER); which makes them a good candidate for document anonymization purposes.

A CRFs is defined as an indirected graph $G = (V, E)$ being V and E the sets of vertices and edges respectively. Vertices represent the components of the random variable $Y = (Y_1, \ldots, Y_n)$ over the label sequence to be assigned and $X = (X_1, \ldots, X_n)$ the random variable over the observed sequence. (X, Y) is a conditional random field if the variables Y_i conditioned to X satisfy the Markov property with respect to their neighbors in the graph.

Linear-chain CRFs

Linear-chains are the simplest form of CRF and can be used to model sequential data, such as tags or labels of words in a sentence. Figure 3.1 shows the general scheme of a CRF applied to sentence tagging.

Let $x_{1:N}$ be a vector of observations and $y_{1:N}$ the associated labels. A linear-chain CRF defines the conditional probability shown in equation 3.1, where Z is a normalization factor (partition function). For each position, we sum over F weighted features (feature functions). Variable λ_i

represents the scalar weight for feature function f_i, and constitute the parameters to be learned by the CRF.

$$p(z_{1:N}|x_{1:N}) = \frac{1}{Z} exp\left(\sum_{n=1}^{N}\sum_{i=1}^{F}\lambda_i f_i(y_{n-1}, z_n, x_n)\right) \tag{3.1}$$

Feature Functions

Feature functions are the key components of CRF. In the case of linear-chain CRF, the general form of a feature function is $f_i(y_{n-1}, z_n, x_n)$, which looks at a pair of adjacent states y_{n-1}, y_n and the input x_n. The feature functions produce a real value. For example, we can define a simple feature function which produces binary values: it is 1 if the current word is *Maria*, and 0 if the current state y_n is PERSON(Equation 3.2).

$$f_1(y_{n-1}, y_n, x_n) = \begin{cases} 1 & \text{if} \quad y_n = \text{PERSON} \quad \text{and} \quad x_n = \text{Maria} \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

In CRF models, we design a set of feature functions to extract features for each word in a sentence. During model training, CRF will try to determine the optimal weights λ_i of the different feature functions that will maximize the likelihood of the labels in the training data.

3.4.2 Implementation

We use the CRF implementation provided by *Python CRFSuite*[2]. Input features are defined as discrete labels with an associated weight between 0 and 1.

The CRF's optimization algorithm updates the weight associated to every feature function. In standard CRF implementations such as *CRFSuite*, input features can only be discrete[3]. Because of that, continuous input variables, such as word-embedding feature vectors have to

[2]Python CRFSuite - Python bindings to CRFSuite [15]

[3]Although there are alternative models such as Continuous Conditional Random Fields that can cope with continuous inputs and outputs.

be previously discretized. One popular way to do so in the field of sentence parsers is to use word clusters instead, as they can be understood as a semantic categorization of words.

Confidence of a labeled sentence

The confidence of a sequence in a CRF can be understood as the probability $P(y|x;w)$, where y is the labeled sequence, x is the input; and w is the array of weights associated to each feature used by the CRF. $P(y|x;w)$ can be computed as in equation 3.3.

$$P(y|x;w) = \frac{\exp(\sum_i \sum_j w_j f_j(y_{i-1}, y_i, x, i))}{\sum_{y' \in Y} \exp(\sum_i \sum_j w_j f_j(y'_{i-1}, y'_i, x, i))} \tag{3.3}$$

Features Considered

For each token of the reports, we use a combination of the features listed below, in a window of up to 7 tokens (3 before and 3 after). If the window is within the beginning or the end of the document, we use the features *Begin of Sentence (BOS)* and *End of Sentence (EOS)* with the offset respect to them. *FreeLing*'s language detection tool is previously applied to every document so that the appropriate lemmatizer and POS tagger are used.

- **Word capitalization.** Each token is assigned a capitalization scheme from the following: *all lowercase*, *all uppercase*, *first uppercase* or *combined*.
- **Is decimal.** Whether or not the token contains numerical characters.
- **Prefixes and suffixes.** All possible suffixes of 3 and 4 characters are considered. If the token is shorter, the prefix and suffixes are padded.
- **Part of speech.** Part of speech of the token determined by *FreeLing*'s POS tagger. Just the most probable assignment is used. Contractions are not splitted, and their POS tags are joined by '+'. E.g. POS-tag of '*del*' is `SP+DA`.

- **Lemma.** The most probable lemma assignment for the token according to *FreeLing*'s lemmatizer.
- **Form.** Original form of the word in the document.
- **Word embedding cluster.** Index of the cluster in which the token can be found. Divisions of 128, 1024 and 8196 clusters are considered for each one of the three word-embedding models: no preprocessing, lowercased or lowercased and with no accents.

In addition to using the clusterings independently, we also allow to use all cluster at once in a single run of the self-learning algorithm. In this case, a portion of the training set reserved to select the best clustering at each iteration. We denote this strategy as $\forall wc$ in chapter 4.

3.5 The Bilinear Long Short-Term Memory Model

Recent work on sentence tagging and sequence modeling use recurrent neural networks, as they can learn patterns in sentences with windows of an arbitrary number of tokens. Long Short-Time Memory Networks are specially interesting, since they use the concept of memory cell to store relevant information of a sentence.

In the case of sequence tagging, layers of this kind of memory units can be stacked and combined with a final CRF output layer [9]. In this final layer, inputs and outputs are directly connected so that neighbor tag information is used for predicting current tags. This has proven to boost accuracy in BIO classification tasks.

We use the Bidirectional Long Short-Term Memory (BiLSTM) network with a CRF output layer architecture used by Dernoncourt et Al. [5] to solve the *i2b2 2014* challenge as a base.

3.5.1 Long Short-Time Memory networks

Long Short-Term Memory (LSTM) networks are a particular kind of artificial Recurrent Neural Networks (RNN) that were first presented in 1997 by Sepp Hochreiter and Juergen Schmidhuber

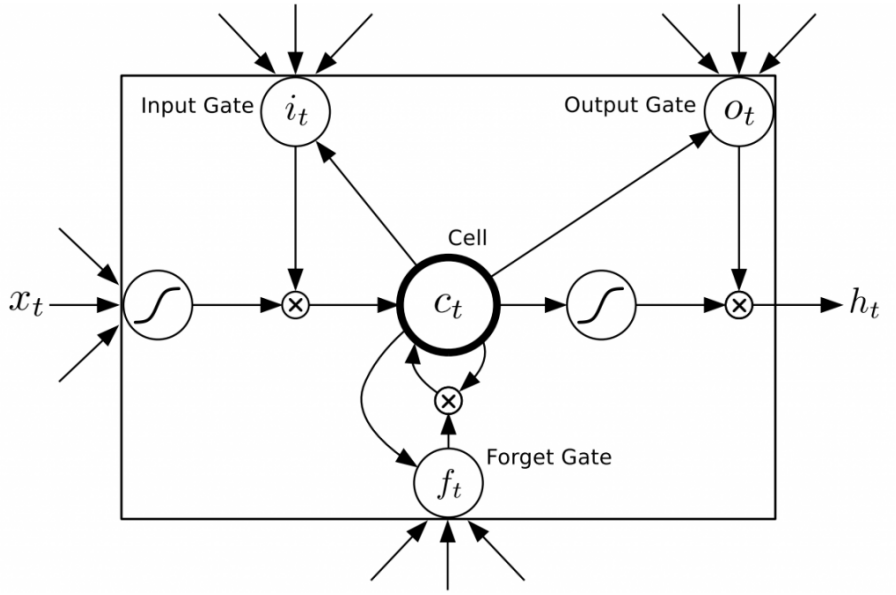


Figure 3.2: Schematic representation of a LSTM cell. Source: LSTM cell schematic (Graves, 2013) - ResearchGate.

[8]. These networks introduce a novel element, the memory units, as a solution to avoid the vanishing gradient problem when RNN are trained using back-propagation through time. LSTM networks help preserve the error that can be back-propagated through time and layers. By doing so, they let RNN to keep learning over many more time steps, thereby being able to model distant recurrences.

LSTMs store the information in a gated cell. This information can be read and written. The cell uses multiple gates, activated by element-wise multiplication by sigmoids, to make decisions about what to store, and when to allow reads. Alternative implementations also include a forget gate, that cleans-up the contents of the memory cell. A diagram of a LSTM cell is shown in Figure 3.2.

LSTM networks have been applied to many fields from time series predictions to handwriting recognition, but they have been specially fruitful in for natural language processing tasks. From Speech recognition, to grammar learning, machine translation and named entity recognition; they have proven to be a very versatile tool for sentence processing tasks.

3.5.2 Implementation

We implement *BiLSTM-CRF* networks is using *Python's Keras* library with *TensorFlow* backend[4]. LSTM layers for both directions use the standard LSTM layer provided by *Keras*, whereas for the output CRF layer we use the implementation in the *Keras-Contrib* extension library[5]. Additionally, we set a dropout factor of 0.5 for regularization. Previous work in the subject has demonstrated that using dropout can improve accuracy when directly using word-embeddings in the input layer.

The network's weights are optimized using the *Adam* first-order gradient-based optimization method. Due to the fact that we want to have an estimation of the marginal probability of the output for each possible tag, we train the model to maximize the product of marginal likelihood over all time steps using categorical cross-entropy as the loss function. Categorical cross-entropy is defined as in equation 3.4. Loss is monitored for determining when to stop training: it is halted when the loss function remains flat for 10 iterations.

$$H(p,q) = -\sum_{\forall x} p(x) \cdot log(q(x)) \Rightarrow L = -y \cdot log(\hat{y}) \tag{3.4}$$

TensorFlow's Recurrent Neural Networks requires all input sequences within a batch to have the exact same length. In order to cope with this limitation, documents are previously divided into sub-sequences of fixed length N'. Sequences are overlapped among them as shown in Figure 3.3 and the output is combined in order not to lose context information. Padding is added at the end of the sequences whenever $len(y_i) < N'$. Special *BOS* and *EOS* tags are used to represent the beginning and end of each document.

Maximum sub-sequence length is a relevant parameter to be assigned. Small values restrict recurrence, as the window of tokens considered by the network is trimmed. Large values may require excessive padding and can make training more complicated. Our experiments use sequences of 24 or 48 tokens, since the average document length is 20 and preliminary tests

[4]Keras: The Python Deep Learning library
[5]keras-contrib : Keras community contributions - GitHub

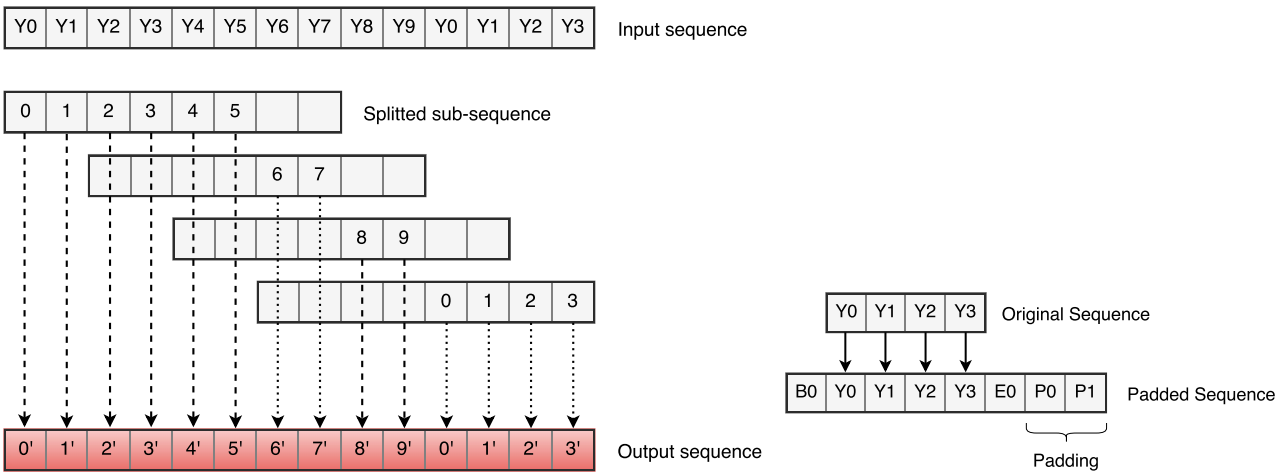


Figure 3.3: Overlapping and padding strategies to keep input sequences' length constant. 25% of the beginning and end of the sub-sequences is overlapped, output is combined so that context before and after are maximized.

showed that longer sequences do not train well due to excessive padding.

The network outputs the probability of each possible tag assignment at any time-step (input token). The most probable tag is finally selected as the argument that maximizes likelihood. The original reports are rebuilt from the sub-sequences and *BOS* and *EOS* as well as padding elements are removed from the network's final output.

Confidence of a labeled document

Computing the probability of a sequence in a recurrent neural network is slightly more complicated. In our implementation, the output layer is fitted to return marginal probabilities on each time-step and optimized via composition likelihood, that is the product of marginal likelihood [3]. The probability of the output at each time-step is conditioned to the output in all the previous time-steps. Consequently, the joint probability of the sequence is defined as in equation 3.5.

$$P(Y^T) = P(y_1) \prod_{t=2}^{T} P(y_t | Y^{t-1}) \quad \text{where} \quad Y^T = \{y_1, y_2, y_3, \ldots, y_T\} \tag{3.5}$$

However, this approach may not be well-suited for our task, as the probability vanishes fast

when the length of the sequence is big and short sequences are likely to be prioritized over long ones. The vanishing probability issue could be solved by applying logarithms, but it would not halve the bias towards short documents with as few named entities as possible.

To circumvent this, we finally opted for defining the confidence of a LSTM network as the minimum output probability of the whole document, as presented in equation 3.6. This definition of sequence confidence only takes into account the time-step with highest uncertainty and ignores the contribution of each output.

$$P'(Y^T) = min\{P(y_t|Y^{t-1})\} \quad \forall y_t \in Y^T \tag{3.6}$$

Network's layout

Figure 3.4 shows the network's layout. Input tokens are first converted to feature vectors by querying the dictionary. Word lookup is implemented as in snippet 2. Feature vector for *BOS*, *EOS* and padding is set to $\overrightarrow{0}$ whereas unknown tokens are set to $\overrightarrow{x}_{mean}$.

Algorithm 2 Pseudo-code of the token lookup method

```
if token ∈ keys(TABLE) then
    return TABLE[token]
else if lowercase(token) ∈ keys(TABLE) then
    return TABLE[lowercase(token)]
else if lemma ∈ keys(TABLE) then
    return TABLE[lemma]
end if
return 0⃗
```

3.6 Observation-based learning

In addition to the aforementioned supervised learning models, we want to find out what can be expected from our self-learning architecture when no labeled training corpus is given as an input. In order to do so, we define gazetteer-based models that can be used as the initial model

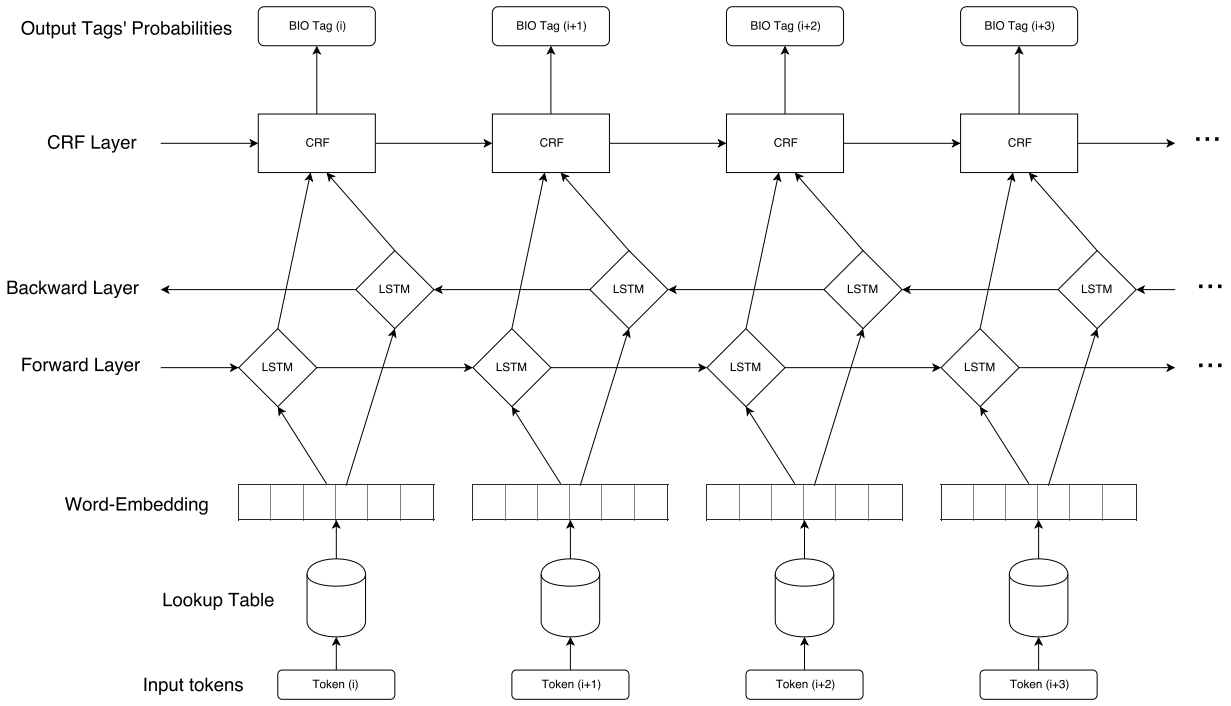


Figure 3.4: Layout of the BiLSTM-CRF network used in this project. LSTM blocks are composed by $\frac{N}{2}$ memory units fully connected to all elements of the feature vector. A dropout factor or 0.5 is applied to both LSTM blocks.

of the self-learning process. This section describes these models and give a brief evaluation of each one of those.

We focus on class PERSON in the observation-based models, since the kind of rules needed for classes PERSON and LOCATION are substantially different and it would require more than twice as much time to implement and combine such models. We will assume that the results obtained for class PERSON are extensible to class LOCATION if the appropriate rule-based model is implemented.

3.6.1 Binary Dictionary Model

The binary dictionary model assigns a BIO tag to each token depending on whether or not the token is found in a certain gazetteer. It uses positive and negative gazetteers, and gives 0 confidence to documents with tokens found in both gazetteers. Documents with no contradictory tokens are assigned a confidence value of 1. When used with frequency dictionaries, a threshold frequency can be used to determine the membership of a token to each one of the gazetteers.

3.6.2 Frequency Divergence Model

The frequency divergence model uses frequency dictionaries as an input, and selects those tokens whose relative frequency in the gazetteer is similar to the relative frequency of the very same token in the whole unlabeled set of documents, ignoring all tokens not in the original gazetteer.

The intuition behind this approach is that the proportions of patients' names in reports should be similar to those in the population. This is a reasonable assumption in a country like Spain, which has free and universal health-care system, and may not be applicable to private health-care institution where surnames' proportions are correlated to families' wealth.

Words whose relative frequency deviates significantly from the gazetteer are probably words with multiple meanings and should not be trusted. The confidence of a medical report can be computed as defined in Equation 3.7, where $div(x_i)$ is the frequency divergence of token i and $div_{th-max}(x_i)$ is divergence threshold. In order to add more flexibility, we define different divergence thresholds depending on the token's capitalization and whether or not the previous token was labeled as PERSON.

$$P(Y) = min(c_i) \quad \forall i \quad \text{where} \quad c_i = 1.0 - min\left(\frac{div(x_i)}{div_{th-max}(x_i)}, 1.0\right) \tag{3.7}$$

3.6.3 Dictionaries

The gazetteer-based models presented in this section are coupled with two different dictionaries: the official count of names and surnames in Catalonia by the *Instituto Nacional de Estadística* (INE) based on census data, and a frequency dictionary of names and surnames provided by the *Institut Universitari d'Investigació en Atenció Primària* (IDIAP).

Table 3.2 shows precision and recall of the dictionary models. In addition to results respect to the full corpus, it also shows values restricted to the confidence threshold used in our experiments. Notice that IDIAP's dictionary performs better in the binary case whereas the generic INE dictionary provides better results with the frequency-divergence model. A possible reason

Model	Dictionary	No threshold		With threshold	
		Precision	Recall	Precision	Recall
Binary	IDIAP	0.114	0.685	0.186	0.571
Binary	INE	0.113	0.564	0.181	0.424
Frequency-Divergence	IDIAP	0.429	0.198	0.5	0.743
Frequency-Divergence	INE	0.636	0.300	0.622	0.848

Table 3.2: Precision and recall of predictions made by the dictionary-based models with and without a confidence threshold. Using strict evaluation.

for this difference is the fact that IDIAP's dictionary contains more terms such as contractions and nicknames, but the population frequency is more reliable in INE's dictionary.

Chapter 4

Experiments and results

4.1 Experimental setup

We apply our framework for different combinations of the proposed supervised learning models and features using different training sets. Evaluation is done using a validation set of 5000 documents. Table 4.1 shows the amount of elements corresponding to each category for the aforementioned training and validation sets. Bootstrapping is applied to health notes issued in November, which are divided in batches of 50000 documents each. Batches are selected in order and are the same for all experiments.

The entities' proportions used by the *select_examples* method in order to ensure that the distribution of the training set resembles the real population is computed using the validation set. Proportions are ensured for both the number of entities and their average length, given that the candidate examples are within the confidence threshold.

	Validation	Training (Limited)	DIP-2.0 (Biased)
Documents	5000	331	1286
PERSON	257	246	239
LOCATION	678	400	550
LOCATION (`ADDRESS`)	7	0	5
LOCATION (`INSTITUTION`)	331	289	290
LOCATION (`GEOGRAPHICAL`)	340	111	255

Table 4.1: Statistics of the datasets used for training and validation

Model	PERSON	LOCATION	Combined
CRF	0.713	0.686	0.696
CRF (*wc*)	0.808	0.723	0.746
BiLSTM-CRF	0.81	0.72	0.746

Table 4.2: Baseline F_1 score for PERSON and LOCATION obtained using 7-fold cross validation of the validation corpus. CRF (*wc*) means that word clusters were used with the CRF model.

We will use the mean F_1 score of the 7-fold cross validation of the best combination of features for the two models using the validation corpus as the baseline, as we did not have at our disposal any alternative large corpus that could be used for this task. Table 4.2 shows the F_1 score achieved with this setup for CRF taggers with morphological features, lemma, POS tag and alternatively word clusters and the BiLSTM-CRF model.

4.2 Results using a small training corpus

The first experiment that we carried out was to apply our method beginning from a very limited training set. The documents were chosen at random among those from December, and it mostly includes documents that contain at least one element to be anonymized. All the supervised learning models described in chapter 3 were considered with the parameters and combinations of feature sets described in the very same section. Table 4.3 shows the F_1 score achieved by the models that performed best both in the supervised system and the semi-supervised one. We see a generalized performance boost using self-learning, specially in those models that performed worst in the initial iteration.

4.3 Results with a biased learning corpus

We have already seen that our self-learning approach can help a supervised algorithm automatically find new patterns previously overlooked due to the limited size of the original training set. Considering this, we argue that our method can be of special help when the original training set is strongly biased towards certain patterns. This is the case for datasets that are obtained

Model	PERSON		LOCATION		Combined		Trend
	Sup.	Semi-Sup.	Sup.	Semi-Sup.	Sup.	Semi-Sup.	
CRF (lm, mph, c)	0.567	0.569	0.364	0.439	0.446	0.488	↗
CRF (lm, mph, pos, c)	0.56	0.569	0.323	0.401	0.419	0.456	↗
CRF (*wc*, lm, mph, pos, c)	0.674	0.696	**0.564**	0.585	**0.604**	0.619	↗
CRF (∀*wc*, lm, mph, pos, c)	0.686	0.778	0.409	0.547	0.505	0.615	↗
CRF (∀*wc*, lm, c)	0.686	0.72	0.42	0.537	0.515	0.592	↗
CRF (*wc*, c)	0.644	0.659	0.518	0.577	0.563	0.603	↗
CRF (∀*wc*, c)	0.702	0.7	0.459	0.549	0.541	0.592	↗
CRF (*wc*)	0.697	0.721	0.438	0.54	0.525	0.593	↗
CRF (∀*wc*)	**0.745**	0.798	0.435	0.552	0.538	0.624	↗
BiLSTM-CRF	0.533	0.785	0.436	0.627	0.494	**0.672**	↗
BiLSTM-CRF (fhu)	0.634	0.667	0.458	0.618	0.531	0.634	↗
BiLSTM-CRF (ls)	0.529	0.744	0.323	0.544	0.423	0.604	↗
BiLSTM-CRF (fd)	0.57	0.763	0.388	**0.629**	0.469	0.67	↗
BiLSTM-CRF (fd, fhu)	0.532	0.683	0.42	0.569	0.486	0.607	↗
BiLSTM-CRF (fd, ls)	0.523	**0.81**	0.333	0.557	0.422	0.633	↗

Table 4.3: F_1 score for PERSONand LOCATIONobtained using a small training corpus. *wc*, lm, mph, pos and c stand for word clusters, lemmas, morphology, POS tags and capitalization respectively. fhu, ls and fd stand for few hidden units, long sequences and few selected documents respectively. Trend indicates whether the combined F_1 tended to decrease or increase using our semi-supervised strategy.

by manually validating examples found using a handcrafted set of rules like regular expressions or gazetteers.

One such set is the DIP-2.0 corpus. This corpus was built by manually labeling documents identified by a set of manual anonymization rules in the form of Augmented Transition Networks (ATN). This rule set is composed of 18 ATN parsers and was only capable of achieving a recall of 0.772 for PERSON and 0.371 for LOCATION, as precision was required to be over 0.5. As a result, the examples in the DIP-2.0 corpus are strongly biased towards the patterns defined by the ATNs, and it would be of special interest to overcome this limitation. Results for the biased corpus are shown in table 4.4.

Evolution of F_1 score for each iteration

Figure 4.1 shows how precision, recall and F_1 score evolved during the different iterations for two executions of the CRF and BiLSTM-CRF models that provided best performance. We see

Model	PERSON		LOCATION		Combined		Trend
	Sup.	Semi-Sup.	Sup.	Semi-Sup.	Sup.	Semi-Sup.	
CRF (lm, mph, c)	0.612	0.597	0.585	0.61	0.597	0.609	≡
CRF (lm, mph, pos, c)	0.615	0.61	0.589	0.601	0.601	0.606	≡
CRF (*wc*, lm, mph, pos, c)	0.721	0.721	0.696	0.696	0.707	0.707	↘
CRF (∀*wc*, lm, mph, pos, c)	0.727	0.731	0.681	0.684	0.698	0.7	≡
CRF (∀*wc*, lm, c)	0.73	0.77	0.686	0.696	0.702	0.718	≡
CRF (*wc*, c)	0.646	0.648	0.669	0.669	0.665	0.665	↘
CRF (∀*wc*, c)	0.696	0.741	0.664	0.665	0.675	0.687	↗
CRF (*wc*)	0.701	0.707	**0.705**	0.712	0.706	0.713	↘
CRF (∀*wc*)	0.744	0.784	0.618	0.709	0.653	0.731	↗
BiLSTM-CRF	**0.784**	**0.852**	0.668	**0.742**	0.702	**0.772**	↗
BiLSTM-CRF (fhu)	0.722	0.822	0.694	0.735	0.707	0.76	↗
BiLSTM-CRF (ls)	0.72	0.826	0.645	0.736	0.666	0.761	↗
BiLSTM-CRF (fd)	0.633	0.819	0.63	0.736	0.635	0.759	↗
BiLSTM-CRF (fhu, fd)	0.753	0.754	0.702	0.721	**0.72**	0.73	≡
BiLSTM-CRF (ls, fd)	0.753	0.825	0.654	0.73	0.684	0.756	↗

Table 4.4: F_1 score for PERSON and LOCATION obtained using the DIP-2.0 (biased) training corpus. *wc*, lm, mph, pos and c stand for word clusters, lemmas, morphology, POS tags and capitalization respectively. fhu, ls and fd stand for few hidden units, long sequences and few selected documents respectively. Trend indicates whether the combined F_1 tended to decrease or increase using our semi-supervised strategy.

that the evolution of BiLSTM-CRF is more erratic even though the overall trend is positive. This highlights the fact that LSTM, as other artificial neural networks, have an associated degree of randomness. Another interesting observation is the drop observed in iteration 1 for the CRF model, which is halved in the following iteration. The model in iteration 0 does not use examples extracted from the unlabeled set, and the addition of them in iteration 1 causes a great drop in performance, which is overcame when more examples are added.

4.4 Comparison with active-learning

Active-learning is a popular semi-supervised learning method that, similarly to our self-learning strategy, uses a previous model to fetch new examples from an unlabeled corpus so that they can be included into the training set. The difference is that the case of active-learning, it is the user or some other information source who validates those documents, hence requiring explicit manual interaction. Given the similarities, it is interesting to see how these two methods stack

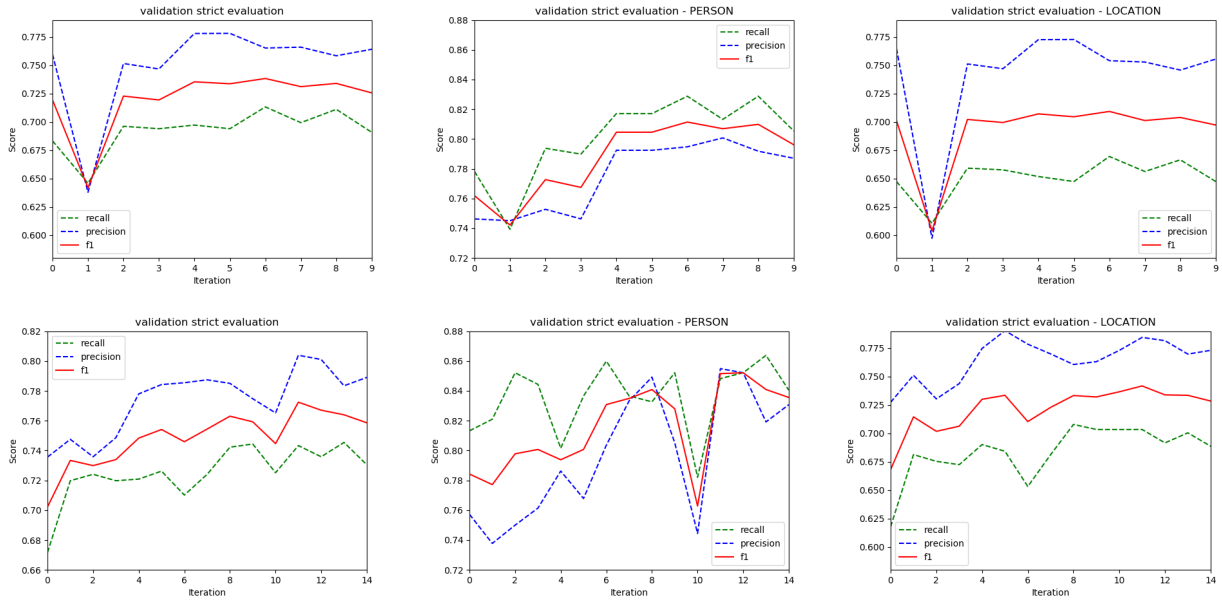


Figure 4.1: Evolution of precision, recall and F_1 score for two different executions of the self-learning algorithm trained with the DIP-2.0 corpus. Top: CRF model. Bottom: BiLSTM-CRF model.

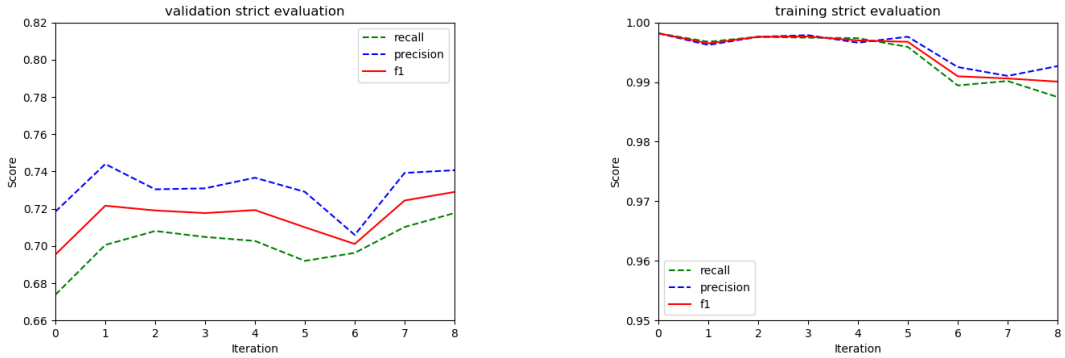
against each other.

Given that active-learning is considerably time-consuming, just a single run composed of 8 bootstrapping iterations was performed. 250 new documents were selected at each iteration among those in which the model learned in the previous step was less confident about, while ensuring that examples of both categories were included.

The supervised learning model used was BiLSTM-CRF with 64 hidden units and subsequences of 24 tokens, since it was the model that yielded the best results in the self-learning experiments. For the same reason, the initial training set was the DIP-2.0 corpus. At the end of the run, the original training set was enlarged with 2000 new documents, almost doubling the initial size.

Contrary to expectations, the final F_1 score achieved with active-learning was worse than with our self-learning approach. Figure 4.2 shows the evolution of the recall, precision and F_1 score in the validation set over the iterations. Even though there is a 0.025 leap in F_1 in the first iteration, F_1 remains flat or even decreases in the successive ones.

There could be multiple reasons for this degradation. But considering the kind documents that we were asked to validate by the system, most of them ambiguous and with severe spelling



(a) Validation (b) Training

Figure 4.2: Combined recall, precision and F_1 score in the validation and test corpus achieved for each active-learning iteration. The number of retrieved documents at each iteration was set to 250 and ranking was inversely proportional to confidence.

errors, the examples added to the training set were not useful in terms of diversity nor they were representative of the most common patterns in health records. This conjecture is supported by the fact that F_1 is strictly decreasing when evaluating the training set, meaning that the model is not able to cope with the ambiguity in the newly added examples.

Our self-learning approach in the other hand, given that the amount of documents added in each iteration is large and statistically representative, is successful in finding new previously overlooked patterns. This suggests that alternative more refined ranking strategies should be adopted for appropriately applying active-learning to the IDIAP corpus, which is outside the scope of this project.

4.5 Results with no manually labeled training corpus

Finally, we see how our self-learning applies to a situation where no labeled training corpus is provided and the only resource we have is a handcrafted observation-based model. The intuition is that, having into account that our system is applicable to biased datasets, it will be able to improve such bad models if it is robust enough against noise and mislabeled examples. Table 4.5 shows the results obtained using the proposed observation-based models for the PERSON category. They show a huge difference in performance when comparing the binary and the

Base	Dictionary	Model	Supervised	Semi-Supervised	Trend
Binary	INE	CRF	0.123	0.123	↘
Binary	INE	BiLSTM-CRF	0.115	0.124	↘
Binary	IDIAP	CRF	0.124	0.133	↘
Binary	IDIAP	BiLSTM-CRF	0.145	0.161	↘
Freq. Div.	INE	CRF	0.527	0.55	↗
Freq. Div.	INE	BiLSTM-CRF	**0.575**	**0.633**	↗
Freq. Div.	IDIAP	CRF	0.526	0.559	↗
Freq. Div.	IDIAP	BiLSTM-CRF	0.523	0.613	↗

Table 4.5: F_1 score for PERSON using handcrafted observation-based model as a base.

frequency-divergence models. This was to be expected, since our system requires an accurate estimation of certainty, which the frequency-divergence model is able to provide but not the binary one.

Chapter 5

Conclusion

5.1 Summary of Thesis Achievements

The results shown in chapter 4 prove that the presented semi-supervised framework is capable of improving traditional supervised anonymization models for all the considered categories. We achieve a performance boost of 0.052 in the combined F_1 score when comparing to the best supervised model.

This improvement is tightly associated to the layer of abstraction and generalization provided by word-embeddings. Models that did not use word-embeddings or word clusters as input features did not improve or even saw performance downgrades due to the added noise. But models that did have them as an input feature, specially those only using word-embeddings, could boost their F_1 score up to a 15%.

We also show how it can help improve strongly biased training sets. For the DIP-2.0 training set, obtained by manually correcting examples retrieved using handcrafted rules, our semi-supervised framework achieved a maximum F_1 score of 0.772, compared to the 0.720 achievable with supervised models. Even when starting from a simple observation-based tagger and no manually labeled training set, our framework could be able to select relevant examples and iteratively improve a supervised model.

5.2 Future Work

Having demonstrated that it is possible to improve state-of-the-art anonymization models by combining a relatively small training corpus with new examples extracted from a large unlabeled set, there is still much work to be done. Improvements could be applied every component of the architecture, from the sequence taggers to the word-embeddings, word clustering or even the document selection algorithm. From all of them, we find specially interesting the improvements listed below:

- Define character-level features to the BiLSTM-CRF model. In our implementation, we only use the *word2vec* feature vector as an input. Character-level could be of great help for identifying tokens that are not in the lookup tables and whose context is not discriminative enough.

- Improve word-embedding representations using regularization techniques such as re-embedding or dropout [17], or alternative word representation models such as GloVe [18].

- Implement an improved instance selection algorithm that is able keep or discard examples based on their performance, inspired by instance-base learning techniques, in order to improve scalability and discard noise.

There is also a considerable room for improvement to the observation-based models. We have shown that our method is capable of improving such weak classifiers without the need of a manually labeled training corpus. However, this improvement depends on the quality of the initial model. We strongly believe that more sophisticated rule-based models combined with our framework could potentially lead to results closer to supervised models for documents with a low density of entities.

Finally, it would be interesting to combine the automatically selected documents using self-learning with manually revised ones, in hybrid active-learning strategy. The goal is to reduce the cost of traditional active-learning, as less manually validated documents would potentially

be needed at each iteration. Moreover, it would prevent the active-learning algorithm from just selecting ambiguous examples, as it was the case in our experiments.

Appendix A

Personal Health Information categories according to the Information Portability and Accountability Act

The Health Information Portability and Accountability Act's guidelines[1] requires all publicly available health records to be freed from information that could be used to identify patients or of relatives, employers, or household members of the individuals. This information can be divided in the following categories:

1. Names and surnames
2. All geographic subdivisions smaller than a state, including street address, city, county, precinct, ZIP code, and their equivalent geocodes, except for the initial three digits of the ZIP code if the population is bigger than 20000 people.
3. All elements of dates (except year) for dates that are directly related to an individual, including birth date, admission date, discharge date, death date, and all ages over 89 and all elements of dates (including year) indicative of such age, except that such ages and elements may be aggregated into a single category of age 90 or older

[1] Guidance Regarding Methods for De-identification of Protected Health Information in Accordance with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule

4. Telephone numbers
5. Vehicle identifiers and serial numbers, including license plate numbers
6. Fax numbers
7. Device identifiers and serial numbers
8. Email addresses
9. Web Universal Resource Locators (URLs)
10. Social security numbers
11. Internet Protocol (IP) addresses
12. Medical record numbers
13. Biometric identifiers, including finger and voice prints
14. Health plan beneficiary numbers
15. Full-face photographs and any comparable images
16. Account numbers
17. Any other unique identifying number, characteristic, or code (with some exceptions)
18. Certificate/license numbers

Appendix B

The corpus of Catalan health records

The corpus of Catalan health records of 2013 is composed by 12 files, each containing the short comments attached to the health records issued during the corresponding month. The first three numbers, divided by vertical bars (/), identify the report. Table B.1 shows some statistics of the health records for the different months.

The documents in this corpus are written in natural language, and usually composed of short sentences lacking verbal phrases or having an incorrect syntactical structure. The words are often misspelled, and capitalization is usually not coherent. There are 582047 different tokens appearing more than 5 times in the whole dataset. If capitalization and accentuation is ignored, the number of different tokens lowers to 413238.

In addition to the aforementioned syntactical and morphological errors, one could make the following observations:

- Use of contractions. An example of this can be seen in the sentence *Pac que finaliza tto*, where the words *Pac* and *tto* are used instead of *Paciente* and *tratamiento*.
- Use of punctuation marks instead of spaces; or lack of them. For example, in the sentence *Algun subcrepitante en bases...Normas.Pulmicort-100 2-1(15 dias).*, the words *bases*, *Normas* and *Pulmicort-100* are not spaced. What is more, in sentence *Controlada HVhebron anualment.*, *HVhebron* should be *H. V. Hebron*.

Table B.1: Statistics of the corpus of Catalan health records of 2013

Month	Reports	# Words	Words/Report
1	2611325	49314471	18.88
2	2857575	54491017	19.07
3	3171363	60351266	19.03
4	2796321	53131386	19.00
5	2841897	53783418	18.93
6	2778917	52900158	19.04
7	2688538	52059565	19.36
8	2082795	42458710	20.39
9	2499976	48533315	19.41
10	2858410	53441475	18.70
11	3037233	58348081	19.21
12	2657986	52207159	19.64
total	32882336	631020021	19.19
test	5000	112281	22.46

- Combination of uppercase and lowercase words and sentences, making capitalization not reliable. An example of this is the document *control enfermeria.CONTROL SINTROM.hoja de monitorizacion..*
- Enumerations of measures and readings from medical analysis. For example, *Usa L/C OD 85º-0.50 +1.00 0.8 /+4.00. OI 115º-1.00 +0.25 0.9 /+3.50.AO 4DP BT en VL.Rx >OD NG. OI NG Ad/3.00.*
- Documents are written in Spanish and Catalan, often combining words and sentences of both languages. This is something common among Catalan speakers.

Appendix C

Examples of PERSON and LOCATION in the IDIAP dataset

Table C.1 shows several instances of the categories PERSON and LOCATION extracted from the IDIAP dataset. Variables that could potentially compromise the patient's privacy have been previously shuffled, ensuring that capitalization and morphological errors are maintained.

Most of the instances of PERSON correspond are either a doctor or a patient, in order to exemplify this, the corresponding subcategory is also indicated in the table; even though this categorization has not been using during evaluation. Similarly, most LOCATIONs are either an address, a geographical location or an institution such as hospitals and schools.

PERSON (DOCTOR)	PC: veure nota anterior . . 2) DErivat desde reumato a DIgestio per estudi de probable mallatia inflamatoria intestinal (colestasi disociada en estudi + rectorragies sese dolor abdominal ni estrenyiment ni a diarrea). . 3) Penndet EMG (Per H discal L5-S1.. amb comentari de si es quirrugica o no) ... i 28 octubre DR **Martí-n**. 4) odontoleg .
PERSON (PATIENT) + LOCATION (INSTITUTION)	primera vista en csmij. llevo a su hermano **gerardo**, ultimamente los padres se han quejado mas del comportamiento de **paco** que de **gerardo** asi que les invito a que traigan a **paco** a consulta. . mc: p de comportamiento, sbt en casa, agresivo, chinchoso, siempre provocando. bajo rend escolar.. la madre sigue refiriendose a su casa como casa de locos. de hecho todos se gritan, elevada tension diaria. posibilidad derivacion **sant pau**?terapia familiar?. la madre es consicente de que trabajan muchas horas y que los niños llaman la atencion pero no parece que haya una idea de disminuir las horas fuera de casa.. la madre tb señala inatencion.
PERSON (PATIENT) + LOCATION (GEOGRAPHICAL)	Ja sempre puja amb ascensor, tot i que és l'únic lloc on practica pq enlloc més no li cal. **Carla** va anar dos dies de colònies, molt bé, però es discutiren amb **Pol**. 'Gelosia' pq ella proposava de quedar-se la nena al juliol un cap de setmana amb mare a **Montgat** i ells anar a **Munich** amb company de grup d'ell. Ell se sentí- atacat, es posí a la defensiva, ella tb. l'atací... però contrarresta la seva ansietat. Ell aleshores 'defensa' que durí a la nena un c. d. s. a soles a **Bèlgica**, però ho diu amb agressivitat.... ella cansada de la situació, veu que es va obrint, comparteix, però molta dificultat. Una exnòvia ja es trobí amb això, que estava atrapat en la culpabilitat, viu a **Dorestad** i als pares ja no els semblava bé.... seguiment. Pdt. nova ferul. la, dolor mandí-bula i cervicals.
PERSON (OTHER) + PERSON (DOCTOR)	Parlo telefonicament amb el marit , avui l'ha visitat l'assistenta social (**Clara**) al domicili , la setmana vinent vindràn a la consulta de la Dra. **Marques**.
LOCATION (INSTITUTION)	ATENDINDA EN URGENCIAS DEL **HOSPITAL SANT** POR DOLOR ABDOMINAL AGUDO.. DIAGNOSTICO AL ALTA GASTROENTERITIS AGUDA.. DIETA LIVIANA. SI EMPEORA EL DOLOR VOLVER A URGENCIAS **SANT PAU**..it..
LOCATION (ADDRESS) + LOCATION (GEOGRAPHICAL)	Control de 3 años. 1º revisión en el centro , hasta ahora en CAP de **Pº S. Juan** hasta los 2 años y el último año en **Tarragona**. Han cambiado de domicilio y corresponde **Clot**. . P- 17 Kg , T-101 cm. Lo cuida la madre , de los 6 meses a los 2 años guarderia. En conjunto come de todo pero le cuesta . Higiene dental . Deposiciones cada dia . No enuresis. Bien vacunado . Onada.

Table C.1: Instances of categories PERSON and LOCATION in the IDIAP dataset. All instances have been previously shuffled.

Bibliography

[1] Mohit Bansal, Kevin Gimpel, and Karen Livescu. Tailoring continuous word representations for dependency parsing. In *ACL (2)*, pages 809–815, 2014.

[2] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[3] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[4] Azad Dehghan, Aleksandar Kovacevic, George Karystianis, John A Keane, and Goran Nenadic. Combining knowledge-and data-driven methods for de-identification of clinical narratives. *Journal of biomedical informatics*, 58:S53–S59, 2015.

[5] Franck Dernoncourt, Ji Young Lee, Ozlem Uzuner, and Peter Szolovits. De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606, 2017.

[6] Sofía Garrido Elustondo, Luisa Cabello Ballesteros, Inés Galende Domínguez, Rosario Riesgo Fuertes, Ricardo Rodríguez Barrientos, and Elena Polentinos Castro. Investigación y protección de datos personales en atención primaria. *Atención Primaria*, 44(3):172–177, 2012.

[7] Yikun Guo, Robert Gaizauskas, Ian Roberts, George Demetriou, Mark Hepple, et al. Identifying personal health information using support vector machines. In *i2b2 workshop on challenges in natural language processing for clinical data*, pages 10–11, 2006.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[10] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[11] Hee-Jin Lee, Yonghui Wu, Yaoyun Zhang, Jun Xu, Hua Xu, and Kirk Roberts. A hybrid approach to automatic de-identification of psychiatric notes. *Journal of Biomedical Informatics*, 2017.

[12] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[13] Ishna Neamatullah, Margaret M Douglass, H Lehman Li-wei, Andrew Reisner, Mauricio Villarroel, William J Long, Peter Szolovits, George B Moody, Roger G Mark, and Gari D Clifford. Automated de-identification of free-text medical records. *BMC medical informatics and decision making*, 8(1):32, 2008.

[14] GS Nelson. Practical implications of sharing data: A primer on data privacy, anonymization, and de-identification—semantic scholar. *Semantic Scholar*, 2015.

[15] Naoaki Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007.

[16] Lluís Padró and Evgeny Stanilovsky. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May 2012. ELRA.

[17] Hao Peng, Lili Mou, Ge Li, Yunchuan Chen, Yangyang Lu, and Zhi Jin. A comparative study on regularization strategies for embedding-based neural networks. *arXiv preprint arXiv:1508.03721*, 2015.

[18] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[19] David Pierce and Claire Cardie. Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 1–9, 2001.

[20] Asif Ekbal Shweta, Sriparna Saha, and Pushpak Bhattacharyya. Deep learning architecture for patient data de-identification in clinical records. *ClinicalNLP 2016*, page 32, 2016.

[21] Amber Stubbs, Michele Filannino, and Özlem Uzuner. De-identification of psychiatric intake records: Overview of 2016 cegs n-grid shared tasks track 1. *Journal of Biomedical Informatics*, 2017.

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

[23] György Szarvas, Richárd Farkas, and Róbert Busa-Fekete. State-of-the-art anonymization of medical records using an iterative machine learning framework. *Journal of the American Medical Informatics Association*, 14(5):574–580, 2007.

[24] Isaac Triguero, Salvador García, and Francisco Herrera. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information Systems*, 42(2):245–284, 2015.

[25] Özlem Uzuner, Yuan Luo, and Peter Szolovits. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563, 2007.

[26] Özlem Uzuner, Tawanda C Sibanda, Yuan Luo, and Peter Szolovits. A de-identifier for medical discharge summaries. *Artificial intelligence in medicine*, 42(1):13–35, 2008.

[27] Bing Xiang and Liang Zhou. Improving twitter sentiment analysis with topic-based mixture modeling and semi-supervised training. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 434–439, 2014.

[28] Hui Yang and Jonathan M Garibaldi. Automatic detection of protected health information from clinic narratives. *Journal of biomedical informatics*, 58:S30–S38, 2015.