applied sciences

MDPI

Article

Named Entity Recognition for Sensitive Data Discovery in Portuguese

Mariana Dias [1,2], João Boné [1,2], João C. Ferreira [1,2,*], Ricardo Ribeiro [2,3] and Rui Maia [1]

1 Inov Inesc Inovação—Instituto De Novas Tecnologias, 1000-029 Lisbon, Portugal; mariana_rebelo@iscte-iul.pt (M.D.); joao.bone@inov.pt (J.B.); Rui.maia@inov.pt (R.M.)
2 ISTAR-IUL, Instituto Universitário de Lisboa (ISCTE-IUL), 1649-026 Lisboa, Portugal; ricardo.ribeiro@iscte-iul.pt
3 INESC-ID Lisboa, 1000-029 Lisbon, Portugal
* Correspondence: Joao.Carlos.Ferreira@iscte-iul.pt

Received: 20 February 2020; Accepted: 20 March 2020; Published: 27 March 2020

Abstract: The process of protecting sensitive data is continually growing and becoming increasingly important, especially as a result of the directives and laws imposed by the European Union. The effort to create automatic systems is continuous, but, in most cases, the processes behind them are still manual or semi-automatic. In this work, we have developed a component that can extract and classify sensitive data, from unstructured text information in European Portuguese. The objective was to create a system that allows organizations to understand their data and comply with legal and security purposes. We studied a hybrid approach to the problem of Named Entity Recognition for the Portuguese language. This approach combines several techniques such as rule-based/lexical-based models, machine learning algorithms, and neural networks. The rule-based and lexical-based approaches were used only for a set of specific classes. For the remaining classes of entities, two statistical models were tested—Conditional Random Fields and Random Forest and, finally, a Bidirectional-LSTM approach as experimented. Regarding the statistical models, we realized that Conditional Random Fields is the one that can obtain the best results, with a f1-score of 65.50%. With the Bi-LSTM approach, we have achieved a result of 83.01%. The corpora used for training and testing were HAREM Golden Collection, SIGARRA News Corpus, and DataSense NER Corpus.

Keywords: sensitive data; general data protection regulation; natural language processing; Portuguese language; named entity recognition

1. Introduction

The amount of sensitive information available on the web, as well as in companies and other industries is growing, which consequently urges for a thriving need to filter and process information, so that it can be used for specific purposes and to be able to protect sensitive information and personal data. The vast majority of existing textual data is unstructured, requiring even more processing efforts to extract reliable information [1]. The emergence of obligations for processing unstructured data has been increasing the focus on the advancement of Named Entity Recognition (NER) [2]. However, for languages with fewer resources, such as the Portuguese language, it is still a challenge and the results are still quite inferior when compared to English, for example. This work strives to evaluate these problems focusing on the research, implementation, and evaluation of NER systems for Portuguese, focusing on Sensitive and Personal Data, with the intent to build a reliable solution that can be used by organizations in a real scenario.

The proposed work aims to transform many of the processes that can be carried out manually and with high cost into automatic processes that can carry out efficiently. It allows organizations to

have confidence in the security of their data and comply with protocols and regulations imposed, as is the case of the General Data Protection Regulation (GDPR) [3]. The main feature of this work is the development of a module based on NLP techniques, focused on named entity recognition for the sensitive data covered by GDPR in unstructured textual documents.

The process of recognizing sensitive data is still a task that is often carried out manually, respecting certain rules, which implies additional time spent and higher chance for errors and failures. Due to this, there has been a great advance in the application of NLP tasks in the real world. However, despite the advances and the encouraging progress in NER, most of the real systems developed base their classification on the document's metadata instead of classifying the content [4]. If we manage to overcome these limitations, the practical applications in other projects besides DataSense would be countless, and to several markets.

The NER Component required the development of a system of sensitive information discovery in text documents. The set of textual information to be processed by the component consists of legal documents, contracts, curricula, minutes, etc. For this reason, this work focused on the study of the NER task for sensitive data, as well as in all involving natural language processing tasks, more specifically in text preprocessing techniques, such as Part-of-Speech Tagging. With this study, we have achieved results that allow us to integrate the developed work and the NER Component into a real-world product, the DataSense Project.

2. Named Entity Recognition

The discovery of Sensitive Data or Personal Information follows different approaches, depending on the challenge and its final goal. Thus, different approaches have emerged to deal with the automatic discovery of sensitive data and information extraction [5]. This detection and classification, in the context of unstructured text information data, is performed using Natural Language Processing techniques, more specifically Named Entity Recognition [6,7]. Named Entity Recognition is a subtask of the Information Extraction (IE) task, in the context of Natural Language Processing (NLP). The purpose of NER is to enable the identification and classification of entities in unstructured text according to a set of predefined categories. Different NLP techniques are applied, which consist of identifying the keywords present in the text and classifying them. The NER task may follow different approaches and also have a very broad set of entity categories. The first approaches appeared in MUC-6 [8], one of the first conferences to introduce the NER task, with the categories: People, places, organizations, time, and numerical expressions. Ever since, this set of categories has been the most common, even though other categories emerged.

Considering existing work, we can see a clear division in the techniques used for NER tasks. The main methods for extracting entities are: Hand-coded techniques and machine learning techniques. Named entity extraction methods based on hand-coded techniques can follow two distinct approaches: Methods based on rules or grammatical patterns and methods based on dictionaries or lexicons. These techniques can obtain good results with no training data [9]. Rule-based methods using grammar rules were the first attempts at solving the NER problem [10]. The biggest disadvantage of ruled-based approaches is that they require a great deal of experience and grammatical knowledge of both the language and the domain. They are also extremely difficult to adapt outside of their scope, and it is hard and expensive to maintain them over time. Another hand-coded technique, also widely used for entity recognition, is the use of dictionaries or word lexicons. These are dependent on previously built knowledge bases in order to extract entities [11]. This knowledge base is usually called gazetteer [12], and its use consists of comparing the words in the text with this gazetteer to find matches. Many of the NER approaches that use a knowledge base resort to Wikipedia [11]. Others use simple stemming and lemmatization techniques to extract more than just exact match words. The use of gazetteers or lexicons is a simple approach to the NER task, however, as previously mentioned, it is always completely dependent on the existence of a previous knowledge base for all entity categories.

After the initial use of hand-coded techniques, by needing to achieve better results, new studies have emerged based on Machine Learning. Supervised learning approaches were first developed by adopting Hidden Markov Models (HMM) [13], as well as Conditional Random Fields (CRF) [14] to train a sequential entity recognizer, both using previously annotated data. The most used are HMM, Maximum Entropy Models, CRF, and some more distinct approaches were that based on Decision Trees models, more specifically Random Forest Models. Recently, approaches based on Deep Learning have emerged, specifically Recurrent Neural Networks (RNN) [15]. These approaches have been consistently growing in the past few years, but the most used model and the one that produces better results is the Long Short-Term Memory (LSTM) or variants of it, such as Bidirectional-LSTM [15]. For Portuguese, the use of LSTM also allowed for better results. In the case of HAREM corpus with an LSTM approach, it is possible to see works with F1-score results close to 80% [16], but far behind the results achieved for the English, that exceed the 92% [17]. This difference in the results for the NER task is quite visible and the existing work for the Portuguese language has much lower results than other languages, which is expected when considering the negative difference in quantity and quality of the available corpora and resources for Portuguese. The same is true for the existing systems, the number of systems for sensitive data discovery developed in the field of NER has increased considerably in recent years [5], but not for Portuguese. The vast majority of systems focus on news and in simple categories rather than sensitive data. In addition, almost all existing systems process Brazilian Portuguese and not European Portuguese [18]. To stand by this statement, we have searched the scientific database Scopus for systems using NER. Out of the 3029 results, only 68 were related to the European Portuguese and the vast majority were focused on English. Further specifying our search query, through a subquery related to the Portuguese language, only 30 documents were significantly related to this language, while another supplementary query to filter only documents mentioning sensitive data, narrowed the results to two. It is worth mentioning that none of these two articles ended up being directly related to applying NER to sensitive European Portuguese data.

3. NER Component

In order to not have a closed environment, a modular architecture was adopted for the development of the NER Component, following a specific processing chain (Figure 1). This architecture allows the different modules that belong to the chain to be configurable and instantiated several times independently.

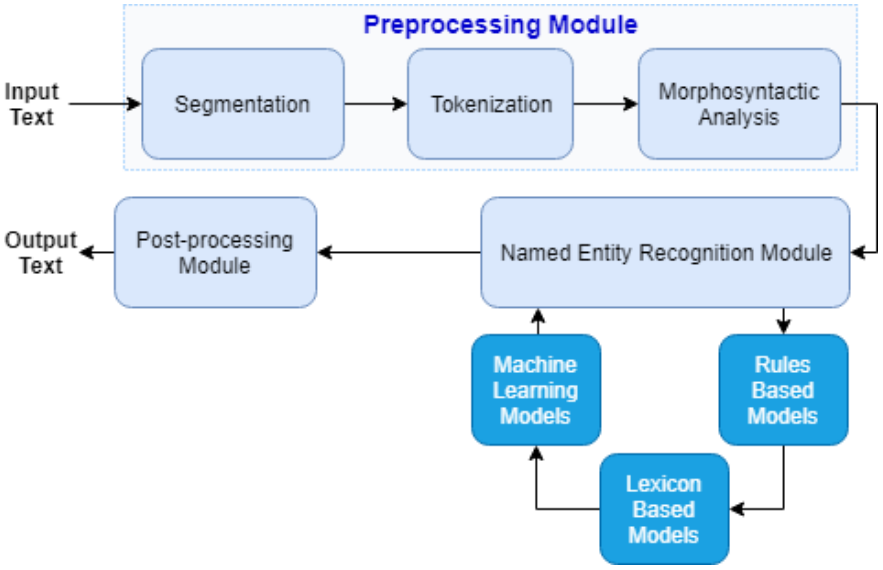

Figure 1. Named Entity Recognition Component chain.

The input of the component consists of text, exclusively in Portuguese, which has previously been treated at the level of images and tables that could be present in a document. Given the input text, the main goal is the Recognition and Classification of sensitive data, considering its class (Personal Identification Number, Socio-Economic Information, etc.). This is done by following the processing chain and all the techniques presented above. The three main steps presented— Preprocessing, Named Entity Recognition, and Postprocessing—consist of the tasks that are divided into modules. Each one of these modules is concerned with solving a specific problem to reach the output text.

3.1. Preprocessing Module

The Preprocessing Module is the first element in the chain. It is responsible for preprocessing and treating the input data, performing a set of preprocessing tasks so that the text can serve as input to the next module. Preprocessing is one of the most important tasks of Natural Language Processing (NLP) and Information Retrieval (IR) studies [19]. Applying this set of techniques to the text means giving it another format so that it can be analyzed, and digested by an algorithm. In this case, the preprocessing module is divided into three parts that are invariably and consecutively executed.

- **Segmentation:** So that each sentence is processed individually without depending on the context of the previous one. We start by dividing the entire text into sentences by the end of sentence punctuation marks: Period (.), question mark (?), exclamation (!), and suspension points (...).
- **Tokenization:** Is performed second on the module chain. This component divides the text in n-grams, words, or sets of words. The number of n-grams can also be parameterized, and this tokenization consists of representing the text as a vector of individual or sets of words. Regarding our approach, some decisions have been made in terms of punctuation, which consist of separating all the nonalphanumeric characters from the words. All punctuation marks except the hyphen (-), the at (@), and the slash (/) are separated by a blank space from all alphanumeric characters in the text. By default, the parameterization used in the processing chain consists of the division into unigrams.
- **Morphosyntactic Analysis:** After tokenization, we perform the morphosyntactic analysis of all separate text in unigrams. The text is analyzed and classified with Part-of-Speech Tagging using different techniques and tools. The task of Part-of-Speech (POS) Tagging consists of analyzing and tagging all the words in the text at the syntactic level. After studying the state-of-the-art, three different implementations were tested and analyzed to select the best to integrate into the NER Component. In the first and second experiments, we used the POS tagging model of the NLTK library. While in the first experiment the model was being applied directly, in the second one we retrained the model with Floresta Sintáctica Corpus [20]. In the third experiment, we used the SpaCy library POS model. After testing, we concluded that the model with the best behavior was the last one, which consists of an implementation based on statistical models and the use of multitask CNN [21]. It achieved an accuracy of 86.4% for the transformed Floresta Sintáctica corpus. Therefore, this is the default model used in the NER Component.

3.2. Named Entity Recognition Module

As we can see in Figure 1, the Named Entity Recognition Module is the second module in the NER component chain. The input of this module consists of the output of the Preprocessing Module, and the output of this module must be the input text annotated in the CoNLL format [22].

It is in the NER module that the models and systems for recognition of sensitive data are implemented. The result produced by this module is the text classified with its respective classes. The classes of entities to recognize in this module were defined accordingly to the sensitive data present in the DataSense project. In this module, we use the division into categories: Personal Identification Number, Socio-Economic Information, etc. Table 1 shows the set of classes of entities considered in this

work, in the column Entities Classes is represented as the name of each of the entities for this work, and in the third column the sensitive data are covered in each class of entity.

Table 1. Classes of entities considered in this work.

Categories	Entities Classes	Sensitive Data Included
Personal Identification Number	NumIdentificacaoCivil	Identification Number
	IdentificacaoBancaria	Bank Identification Number
	NumCartaoDeCredito	Credit Card Number
	NumIdentificacaoFiscal	Tax Identification Number
	NumPassaport	Passport Number
	NumSegSocial	Social Security Number
	NumUtenteDeSaude	National Health Number
	ContactoTelefonico	Telephone Number
	NumCartaConducao	Driving License Number
Socio-Economic Information	Pessoa	Person Names
	Local	Addresses, Locals
	Organizacao	Organizations
	Tempo	Dates
	Valor	Values, Ordered values
	Med	Medical data
	Profissao	Jobs, Professions
Other	CodigoPostal	Postal Code
	EnderecoEletronico	E-mail address

The recognition of named entities is based on three different submodules: Rule-Based Models, Lexicon-Based Models, and Machine Learning Models.

1. **Rule-Based Models:** Several Information Extraction and Named Entity Recognition approaches are based on rules. This first component of the NER Module implements different rule-based models to discover some of the entity classes. The entity classes that are discovered at this stage of the component chain are all associated with sensitive data related to the Personal Identification Numbers category, including postal codes, email addresses, and some date formats. In addition to these rules there is, in some cases, an extra validation. This validation is performed on all personal numbers in which there is a control validation, check digit, or checksum. It allows us to disambiguate and have a greater certainty of cases such as the telephone number and the tax identification number, both containing nine digits. For the telephone number, for example, a set of rules and also a set of context words were created. The regex used for the extraction of this entity was: 9[1236][0-9]{7}|2[1-9]{1}[0-9]{7}|1[0-9]{2,3}, while the list of context words for it was: **Contacto, contato, telemovel, telefonico, telefónico, contactar, fax**. It consists of a set of Portuguese words that usually appear in documents related to telephone contacts. Another feature of this implementation is the context, which has been added to the model in order to solve errors in some of the data types, mainly those of the Personal Identification Number category. The context consists of a specific word or set of words for each class of entity that must exist in the text in order to confirm the result achieved with the rules.
2. **Lexicon-Based Models:** Is the second component in the processing chain. This approach was chosen due to the lack of Portuguese corpus classified for the task of Named Entity Recognition and the good results often achieved with this type of approaches [23]. These lexicon-based models combine the results of morphological analysis, a set of lexicons, and techniques of stemming and lemmatization. The goal is the recognition of the entity classes: PESSOA, LOCAL, PROFISSAO, MED, VALOR, and TEMPO. For each entity, we used different lexicons with their own specific characteristics. This type of implementation consists of comparing the tokens present in the text with the lexicon, and understanding if they correspond to the

same entity. The first entity class is PESSOA, which corresponds to the names of people. For this implementation two different lexicons were used in order to catch both female and male names, and these lexicons can be obtained from the Public Administration Data Portal (https://dados.gov.pt/pt/datasets/nomesfeminino). The LOCAL entity follows the same implementation used for the names above, as two lexicons were also used in this case, and each entry may correspond to more than one word, as is the case of 'United Kingdom'. The first lexicon, with more than 18,000 entries corresponds to the set of all Portuguese cities, municipalities, and parishes, available in Gov Data (https://dados.gov.pt/pt/datasets). For the entities PROFISSAO and MED we have also used the comparison with lexicons, but using a different approach. The two lexicons are from Wikipedia (https://pt.wikipedia.org) information. The entity VALOR should extract all existing values from the text, which may correspond to the value of a contract, a fine, etc. In this type of entity, the value can be written both numerically and in full, and to cover both cases, we used Part-Of-Speech (POS) Tagging classification. For the words or symbols that come associated with the values, besides the use of the tag 'SYM' of POS Tagging, a lexicon has been created with the most relevant words that should be considered. This lexicon consists of words such as 'dollar', 'euro', 'millions', etc. The last entity implemented was TEMPO, and to be able to deal with it, a lexicon was created with all the months in Portuguese and English, as well as their abbreviated forms. Some of these entities were also implemented with Machine Learning models, with the goal of understanding how to achieve the best results for each class of sensitive data.

3. **Machine Learning Models:** Is the last subcomponent on the chain of the NER module. We conclude from the current state-of-the-art analysis that for the most ambiguous entities and for those in which there are no well-defined rules, the best results are achieved through machine learning methods and, out of these, the most recent approaches are based on the study of neural networks. In the next section, we present the Machine Learning approaches for NER used in this work. These approaches were carried out for a smaller set of entities: PESSOA, LOCAL, TEMPO, VALOR, and ORGANIZACAO; For these entities and in this experiment, we had two different approaches:

    - We implemented the two statistical models most commonly used in the tasks of Named Entity Recognition, Conditional Random Field, and Random Forest;
    - We studied a neural network approach, in which a Bidirectional-LSTM was chosen for the different approaches implemented, and the used corpora were **HAREM golden Collection** [24] and **SIGARRA News Corpus** [25].

3.2.1. Statistical Models

NER approaches with statistical models typically require a large amount of training data, and these have not been used as much to avoid the overwhelming annotation effort [26,27]. Since we only have two not very extensive corpora, this difficulty can be overcome. The models chosen for this approach were a Conditional Random Fields Model (CRF) and a Random Forest Model (RF).

Conditional Random Fields: The CRF model implemented in this work is based on the implementation of Korobov M. and Lopuhin K., for the corpus CoNLL2002, available at GitHub (https://github.com/TeamHG-Memex/eli5/blob/master/notebooks/sklearn-crfsuite.ipynb), as well as the implementation of the NERP-CRF system [28]. The tasks performed to define and extract features were based on the work of McCallum A. and Li W. [29].

The implemented model is a nondirectional graphical model used to calculate the conditional probability of the output nodes' values based on the values assigned to the corresponding input nodes. This model makes a first-order Markov independence assumption, so it can be understood as a conditionally trained finite state machine. The model has $x = (x_1, \ldots, x_m)$ as input sequence, where x represents the set of ordered words belonging to a sentence. We have $y = (y_1, \ldots, y_m)$ as the

output sequence states that corresponds to the classes of named entities, where y is a set of Finite State Machines (FSM) corresponding to entity classes that match x. We model the conditional probability through $p(y_1, \ldots, y_m|x_1, \ldots, x_m)$. CRFs define a conditional probability of an output state given an input sequence, by the Hammersley-Clifford theorem [30]:

$$P_\Lambda(y|x) = \frac{1}{Z_x} \exp\left(\sum_{m=1}^{M} \sum_{k} \lambda_k f_k(y_m - 1, y_m, x, m)\right) \quad (1)$$

where Z_x is a normalization factor for all state sequences, λk is a learned weight for each feature function, and $f_k(y_{m-1}, y_m, x, m)$ is an arbitrary feature function over its arguments. The feature function has been set to have a value of 0 in most cases and to have a value of 1 if y_{m-1} is the state 1, where 0 corresponds to the tag 'O' and 1 corresponds to the tag 'PERSON'. The feature function can access the entire input sequence, including queries on previous and next words, so $f_k(\cdot)$ can range between$-\alpha \ldots +\alpha$. The features, f_k, are based on the set of features used. In this implementation, the POS tags can be seen as pre-extracted features, but more features were extracted, such as:

- Parts of words through stemming;
- Simplified POS tags;
- Confirmation of capital letters, lower case letters, titles, and digits;
- Resources from nearby words.

To consider the effect of adding a new feature, a new sequence template is defined with an additional feature, g, with weight μ.

$$P_\Lambda + g, \mu(y|x) = \frac{P_\Lambda(y|x) \exp\left(\sum_{m-1}^{M} \mu g(y_m - 1, y_m, x, m)\right)}{Z_x(\Lambda, g, \mu)} \quad (2)$$

By converting the corpus to a dictionary list format, with the tokens and all associated features, we were able to train and test the CRF.

Random Forest: The second statistical model implemented in this work, Random Forest model is a machine learning algorithm that works through decision trees. The model is trained to create a group of decision trees with a random subset of the data. The implementation carried out follows the implementation of Shoumik available at Kaggle (https://www.kaggle.com/shoumikgoswami/ner-using-random-forest-and-crf) and the approach of feature extraction of Jin N. [31]. In terms of features, we tried to bring the model as close as possible to the previous one, in order to compare them. The implemented model is a simple tree-based classification model, that consists of a large number of deep trees, where each tree is trained using a random selection of features [32], so as to gain a complete understanding of the decision-making process. Each tree takes a path (or paths) from the tree root to the leaf, consisting of a series of decisions, held by a particular feature, each of which contributes to the final predictions. In this case, the model with M leaves divides the feature space into M regions, R_m, $1 \leq m \leq M$. Additionally, the tree prediction function is then defined by:

$$f(x) = \sum_{m=1}^{M} c_m I(x, R_m) \quad (3)$$

where M is the number of leaves in the tree, R_m is a region in the space of the features corresponding to leaf m, c_m is a constant corresponding to region m, and finally I is the indicator function. The indicator function returns to 1 if $x \epsilon R_m$ and 0 if not. The value of c_m is determined in the training phase of the tree and R_m represents the extracted features, which correspond to the same features in the previous model.

Before training the Random Forest model, we converted the data into a simple feature vector for each word. That is, each vector consists of the word and the set of features used in this model.

3.2.2. Neural Network Model

After the literature review, we noticed that the most used approaches, and also the ones that produce better results, have been using LSTM (Long Short-Term Memory) [33].

For this experiment, the SOTA algorithm [34] was implemented following the approach of Chiu J. and Nichols E. [17]. The implementation is based on a Bidirectional-LSTM (Bi-LSTM) [35], and it also uses a Convolutional Neural Network (CNN) to identify character-level patterns. The LSTM cells are the building block of Recurrent Neural Networks (RNNs). While plain LSTM cells in a feedforward neural network process text from left to right, Bi-LSTMs also consider the opposite direction, which allows the model to discover more patterns. In this case, the model not only considers the sequence of tokens after a token of interest but also before the token of interest.

For this implementation, an embedding representation was used for each word. All words were mapped to vectors through the embeddings provided by fastText [36]. This means that all words and characters were mapped to real numbers that the neural network can work with. All words, except the already removed stopwords, were mapped using the pretrained Portuguese dictionary of fastText. At the model architecture level, the Bi-LSTM layer forms the core of the network and is composed of three entries:

- Character-level patterns are identified by a convolutional neural network;
- Word-level input from fastText embeddings;
- Casing input (whether words are lower case, upper case, etc.).

After training the model, the softmax activation layer generates the final outputs.

3.3. Postprocessing Module

Postprocessing is the last module of the Named Entity Recognition component chain (Figure 1). This module is meant to treat the results achieved from the previous NER modules and return the text (output), and the entities found with the desired format to the user. It allows the user to choose to view the result in five different ways, since there are different types of outputs that can be shown, depending on their preference.

4. Experiments and Results

As we saw before, different methodologies were applied to the NER task: Ruled-Based Models, Lexicon-Based Models, and Machine Learning Models. However, all experiments and tests performed were carried out on a single machine, using Python Language. As a consequence of not having any corpus for evaluation that contains all the classes of entities that were worked on in this work, the different techniques were used for different classes of entities. The datasets are crucial for the success of any Machine Learning work, but the NER task for the Portuguese language presents several problems due to the lack of training and testing datasets. The only freely available Portuguese dataset annotated with classes of entities was the one developed for the HAREM events [24]. One other Portuguese dataset is the SIGARRA News Corpus, annotated for named entities, consisting of a set of 905 news manually annotated (https://hdl.handle.net/10216/106094), which was taken from the SIGARRA information system at the University of Porto (https://sigarra.up.pt). From these two datasets, none presents all the classes of entities used in this work, and mainly in the two corpora, neither of them respects its context, seeing that the majority of the documents were either News or Web pages' text. For this reason, one of the key aspects of this work was the construction of the test corpus, the DataSense NER Corpus. This corpus was built with the aim of understanding the results obtained when applied to the real context of the DataSense Project.

All classes discovered through Rule-Based models, as well as the Profissao (Job) and Med (Medical data) classes can only be evaluated by the DataSense NER Corpus. The class Valor (Value), which is present in both Lexicon-Based and Machine Learning methods used, was only trained with

and analyzed on the corpus HAREM, since the corpus SIGARRA does not have this entity class annotated. All other classes of entities could be evaluated with both the HAREM and SIGARRA corpora, which allowed for a comparison between the results obtained. The evaluation of the task of Named Entity Recognition is based on the metrics: Precision, recall, and f1-score. Table 2 shows the results of all the tests performed, while the experiments and results are detailed individually in the following subsections.

Table 2. Models results.

Model	Metrics	HAREM Golden Collection	SIGARRA News Corpus
Lexicon-Based	Precision	71.00%	51.32%
	Recall	55.60%	74.10%
	F1-score	62.36%	60.64%
Conditional Random Fields	Precision	63.48%	73.60%
	Recall	44.35%	59.01%
	F1-score	52.21%	65.50%
Random Forest	Precision	49.87%	65.8%
	Recall	36.12%	50.1%
	F1-score	41.89%	56.89%
Bidirectional LSTM	Precision	-	81.13%
	Recall	-	75.61%
	F1-score	-	78.27%

4.1. Lexicon-Based Models Evaluation

The recognition of Named Entities (NE) based on lexicons was one of the methods used in this work, mainly for the classes of entities for which there is no annotated corpus. However, since these experiments were also performed with lexicons for the classes Pessoa (Person), Local, Tempo (Time), and Valor (Value), they were evaluated using the corpus HAREM and SIGARRA. In this case, the totality of the two corpora were used for the evaluation. The obtained results are represented in the first row of Table 2.

We cannot draw proper conclusions, as there is no explicit difference between the two corpora. However, analyzing detailed f1-score results for each class of entities, allows us to conclude that the classes of entities PESSOA and LOCAL cannot be used to achieve satisfactory results. This is due to the inexistence of many names and places in the lexicons used, as well as the fact that there is a great deal of confusion between the two entities. Comparing the results obtained to other works with the same lexicon-based approach and with the same corpus, we can draw some conclusions. For the HAREM Golden Collection [24], the results obtained for class entities PESSOA and LOCAL are very close to those obtained with the REMMA system [37], but when comparing the same system for TEMPO and VALOR classes [24], we get results with an f1-score 20% higher on average. Another system with a similar approach that used the same corpus is Rembrandt [38], this system got its results for the TEMPO class with an f1-score of 33.07%, much lower than ours. For the remaining classes of entities, the results are similar, except for the PESSOA class where the Rembrandt system achieves results of 47.40%, slightly higher than ours. In terms of the use of lexicon-based approaches, we were able to outperform the existing state-of-the-art results for the class TEMPO, maintaining the results for the remaining classes of entities. For the SIGARRA corpus, there is only one work [38], presenting a proposal based on NER tools, which in this case achieves higher average results than those obtained with lexical-based methodologies.

4.2. Statistical Models Evaluation

The corpus HAREM and SIGARRA were used, in this case for the training and testing of both models. To perform the evaluation, we used 5-fold cross-validation as an input parameter for the

classifier, that is, we divided the corpus into five subsets, and the model was trained and tested on them. In the second and third rows of Table 2 we can see the results of our statistical models.

By analyzing the results, we concluded that the results obtained for the HAREM Golden Collection corpus are lower and that this corpus is not enough to train a model and have satisfactory results. Still, we conclude that the Conditional Random Fields model achieves better results when compared to the Random Forest Model. The results obtained with the CRF model and the HAREM Golden Collection were compared to the results of two NER systems, which conducted experiments under the same conditions of this study. The first NERP-CRF system [39] obtained lower results but using the total set of categories available in the corpus. The results of f1-score were 51.57%, 5% lower than the results we obtained in this experiment. Another system that uses the same CRF model is the CRF+LG [40], this system with the use of the CRF model obtained results of 65.33%, higher than the results we obtained. This is due to the use of gazetteers that support the model classification. On the other hand, when comparing the results obtained by both models with the same CRF and Random Forest models applied to the English language, the results obtained with the corpora HAREM and SIGARRA have an f1-score 10% lower on average [41].

4.3. Neural Network Model Evaluation

The final experiment was the implementation of a Bi-LSTM. This model, unlike the others, was trained and tested only with the SIGARRA News Corpus, this is due to the insufficient number of samples in the other corpus to train the model. The corpus was previously divided into three sets: Training, development, and testing. The embed function that creates word-level embeddings was used to generate an embedding representation for each word of the text. The parameters used for training the model were: 80 epochs, 0.68 dropouts, 275 LSTM state size, and three convolutional widths. After training the model and generating the final outputs through the softmax layer, in IOB tagging format, it was possible to perform the evaluation of the model presented in the last row of Table 2.

We can see that the results obtained by this model are higher than those obtained by the statistical models. This model obtained an f1-score of 78.25%, about 13% higher compared to the best statistical model implemented, for the same corpus. In an attempt to compare the state-of-the-art models to the current one tested, we did not find an approach for the same corpus, but we were able to understand that the same Bi-LSTM model applied to the English language has results 12% higher on average [16], [17]. A similar approach with an LSMT-CRF model [16], for a corpus in Portuguese, presents f1-score results of 76.03%, lower than the 78.27% we achieved with this model. By analyzing similar models and improving results with the statistical models tested, we were able to understand that with a larger corpus the results with this type of model greatly improve [17].

Finally, it is possible to see, for each corpus tested, the comparison results by entity class for each approach. The results achieved with the SIGARRA News Corpus were significantly better in all experiments than the results achieved with the HAREM Golden Collection. This was due to the fact that the SIGARRA corpus is larger than the HAREM, which improves the training process. In addition, the SIGARRA News Corpus contains many documents with the same structure, making it easier to learn.

In Table 3, we can see the average of the results for each entity class in each model. The best results achieved for each class are represented in bold on the table, and we can see that the best results for the entities TEMPO and VALOR were achieved with the Lexicon-based models and, for the remaining entities, the best results were achieved by the Bi-LSTM model.

Table 3. Evaluation results by entity class by model.

Entity Class	Lexicon-Based Model	CRF Model	RF Model	Bi-LSTM Model
TEMPO	**91.7%**	65.9%	75.2%	71.27%
VALOR	**62.8%**	34.6%	18.6%	-
PESSOA	39.5%	69.4%	60.2%	**80.78%**
LOCAL	51.9%	77.5%	58.4%	**80%**
ORGANIZACAO	-	47.1%	34.1%	**80.5%**

4.4. Named Entity Recognition Component Validation

The last evaluation performed for this work consisted of the validation of the Named Entity Recognition Component development. Its main goal was to assess both the performance of the NER component and the quality of the recognition of sensitive entities. For this evaluation we used the DataSense NER Corpus, previously annotated. Before performing the evaluation tests, there was a set of decisions that were made in order to understand which models would be part of the NER Component. For the Named Entity Recognition Module, and considering the results of f1-score and performance for each of the tests, a set of different tasks was chosen in order to cover all classes of entities required for the DataSense Project and the best results for each entity. In Table 4, we can see the named entity recognition methods chosen for each class of entities.

Table 4. Method of Named Entity Recognition (NER) used by a class of entity for NER Component.

Entity Classes	Used Models
Personal Identifications Numbers CodigoPostal EnderecoElectronico	Rule-Based Models
Profissao Med Tempo Valor	Lexicon-Based Models
Pessoa Local Organizacao	Machine Learning Models (Bi-LSTM)

The evaluation of DataSense Ner Corpus was performed with this set of models. Its results are presented in Table 5.

Table 5. DataSense NER Corpus evaluation results.

Metrics	Results
Precision	87.60%
Recall	79.02%
F1-score	83.01%

The NER Component had an f1-score of 83.01% in the DataSense NER Corpus and took 1716 s to complete the processing of all 78 documents. In Figure 2, we can see the detailed analysis of the f1-score results for each class of entities.

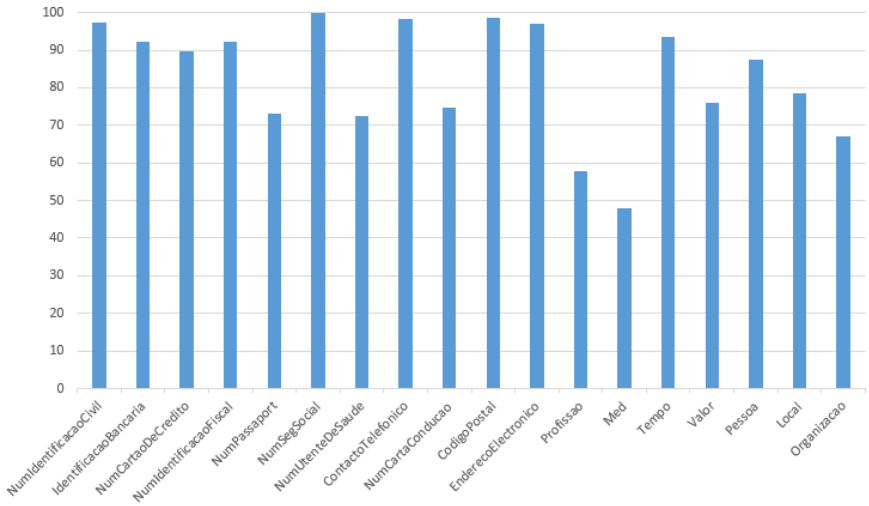

Figure 2. NER Component f1-score results by class, where the x-axis corresponds to the classes of entities and the y-axis to the f1-score results obtained.

From the Figure 2, we can conclude that there are some classes, such as Profissao (Job) and Med (Medical data), which have much lower results when compared to the other classes, because these two classes have been implemented with lexicon-based models and these lexicons are very small and very lacking in these two entities. However, in general, all classes presented good results.

5. Conclusions

The main goal of this work was to develop a functional prototype of Named Entity Recognition for the Portuguese language. The focus of the developed prototype was the recognition of sensitive data in unstructured texts, according to all categories covered by GDPR. This prototype was validated, under the Portugal 2020 DataSense Project, through the tests of efficiency and performance. This validation was carried out with the project stakeholders, achieving an f1-score of 83.01% in the NER task.

The work was developed using a hybrid approach, and several experiments were done in order to achieve the best results for each entity class. A rule-based model and morphological analysis were implemented, achieving the best results for entities with well-defined formats and that follow strict rules. Models based on lexicons were also implemented for a reduced set of entities, achieving an f1-score result of 62.36% for HAREM and 60.64% for SIGARRA. Although the global results when using lexicon-based models are lower than the current state-of-the-art, for TEMPO and VALOR entities the results were higher than those achieved with other methodologies, and they were a way of solving the PROFISSAO and MED entities for which there was no labelled data in Portuguese, but were necessary for the proposed prototype. These two approaches have, however, some limitations. They are restricted only to a fixed set of entities and well-defined rules. They are, nevertheless, the ones that have achieved the best results and a greater confidence in the results obtained. For the remaining classes of entities, different experiments were carried out, including the implementation of statistical machine learning models and the implementation of a Bidirectional-LSTM neural network. The two statistical models—CRF and RF—allowed us to conclude that the first one achieved better results than the second. With these two models, we were also able to understand that the HAREM corpus is not enough for training more complex models, due to its size and the reduced number of annotated entities. Finally, the third implemented model was the Bidirectional-LSTM, and ended up obtaining the best f1-score results in NER and thus, was used in the prototype. The use of different methodologies covered all sets of entities that represent sensitive data. We also conclude that it is possible for the Portuguese language to have valid results for named entities recognition tasks, and it can be used in real scenarios with a remarkable value in the Portuguese market.

Author Contributions: This article was developed within the DataSense Project to which M.D. belongs as developer and R.M. as project manager. The results presented and the experiments performed were developed within the scope of M.D.'s Master's thesis and with the scientific help of its supervisor J.C.F. and R.R. Coorientador. The article was written by M.D. with the help of J.B., who currently continues the work presented here. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the DataSense Project. This project is being led by Link Consulting and received cofunding from the FEDER—Lisbon 2020, PT 2020, European Union's PT 2020 research, and innovation program under grant agreement cod POCI-01-0247-FEDER-038539.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dale, R.; Moisl, H.; Somers, H. (Eds.) *Handbook of Natural Language Processing*; CRC Press: Boca Raton, FL, USA, 2000.
2. Nadeau, D.; Sekine, S. A survey of named entity recognition and classification. *Lingvist. Investig.* **2007**, *30*, 3–26.
3. Dias, M.; Maia, R.; Ferreira, J.; Ribeiro, R.; Martins, A. DataSense Platform. In Proceedings of the IASTEM—586th International Conference on Science Technology and Management (ICSTM), Bandar Seri Begawan, Brunei, 11–12 April 2019.
4. Clough, P. Extracting metadata for spatially-aware information retrieval on the internet. In Proceedings of the 2005 Workshop on Geographic Information Retrieval, Bremen, Germany, 4 November 2005; pp. 25–30.
5. Korba, L.; Wang, Y.; Geng, L.; Song, R.; Yee, G.; Patrick, A.S.; Buffett, S.; Liu, H.; You, Y. Private data discovery for privacy compliance in collaborative environments. In *International Conference on Cooperative Design, Visualization and Engineering*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 142–150.
6. Cardie, C. Empirical methods in information extraction. *AI Mag.* **1997**, *18*, 65.
7. Ciravegna, F. 2, an adaptive algorithm for information extraction from web-related texts. In Proceedings of the IJCAI—2001 Workshop on Adaptive Text Extraction and Mining, Seattle, WA, USA, 4–10 August 2001.
8. Grishman, R.; Sundheim, B. Design of the MUC-6 evaluation. In Proceedings of the 6th Conference on Message Understanding, Columbia, MD, USA, 6–8 November 1995; Association for Computational Linguistics: Stroudsburg, PA, USA, 1995; pp. 1–11.
9. Brill, E.; Mooney, R.J. An overview of empirical natural language processing. *AI Mag.* **1997**, *18*, 13.
10. Mikheev, A.; Moens, M.; Grover, C. Named entity recognition without gazetteers. In Proceedings of the Ninth Conference on European chapter of the Association for Computational Linguistics, Bergen, Norway, 8–12 June 1999; Association for Computational Linguistics: Stroudsburg, PA, USA, 1999; pp. 1–8.
11. Gattani, A.; Lamba, D.S.; Garera, N.; Tiwari, M.; Chai, X.; Das, S.; Subramaniam, S.; Rajaraman, A.; Harinarayan, V.; Doan, A. Entity extraction, linking, classification, and tagging for social media: A wikipedia-based approach. *Proc. VLDB Endow.* **2013**, *6*, 1126–1137. [CrossRef]
12. Torisawa, K. Inducing gazetteers for named entity recognition by large-scale clustering of dependency relations. In Proceedings of the ACL-08: HLT, Columbus, OH, USA, 15–20 June 2008; pp. 407–415.
13. Zhou, G.; Su, J. Named entity recognition using an HMM-based chunk tagger. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; pp. 473–480.
14. Finkel, J.R.; Grenager, T.; Manning, C. Incorporating non-local information into information extraction systems by gibbs sampling. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Ann Arbor, MI, USA, 25–30 June 2005; Association for Computational Linguistics: Stroudsburg, PA, USA, 2005; pp. 363–370.
15. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. *arXiv* **2016**, arXiv:1603.01360.
16. De Castro, P.V.Q.; da Silva, N.F.F.; da Silva Soares, A. Portuguese Named Entity Recognition Using LSTM-CRF. In *International Conference on Computational, Processing of the Portuguese Language*; Springer: Cham, Switzerland, 2018; pp. 83–92.
17. Chiu, J.P.; Nichols, E. Named entity recognition with bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Linguist.* **2016**, *4*, 357–370. [CrossRef]

18. Mota, C.; Santos, D.; Ranchhod, E. Avaliação de reconhecimento de entidades mencionadas: Princípio de AREM. In Proceedings of the Avaliação Conjunta: Um Novo Paradigma no Processamento Computacional da língua Portuguesa, Computational Processing of the Portuguese Language: 7th International Workshop. PROPOR, Itatiaia, Brazil, 13–17 May 2006; pp. 161–175.
19. Kannan, S.; Gurusamy, V. Preprocessing Techniques for Text Mining. *Int. J. Comput. Sci. Commun. Networks* **2014**, *5*, 7–16.
20. Afonso, S.; Bick, E.; Haber, R.; Santos, D. Florestasintá(c)tica: A treebank for Portuguese. In quot. In *Manuel González Rodrigues, Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002), Las Palmas de Gran Canaria, Spain, 29–31 May, 2002*; Paz Suarez Araujo, C., Ed.; ELRA: Paris, France, 2002.
21. Arnold, T. A tidy data model for natural language processing using cleannlp. *arXiv* **2017**, arXiv:1703.09570. [CrossRef]
22. Ramshaw, L.A.; Marcus, M.P. Text chunking using transformation-based learning. In *Natural Language Processing Using Very Large Corpora*; Springer: Dordrecht, The Netherlands, 1999; pp. 157–176.
23. Ratinov, L.; Roth, D. Design challenges and misconceptions in named entity recognition. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning, Boulder, CO, USA, 4 June 2009; Association for Computational Linguistics: Stroudsburg, PA, USA, 2009; pp. 147–155.
24. Santos, D.; Cardoso, N. A golden resource for named entity recognition in Portuguese. In *International Workshop on Computational, Processing of the Portuguese Language*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 69–79.
25. Môro, D.K. *Reconhecimento de Entidades Nomeadas em Documentos de Língua Portuguesa*; TCC- Universidade Federal de Santa Catarina Araranguá, Tecnologias de Informação e Comunicação: Florianópolis, Brazil, 2018.
26. Lin, D.; Wu, X. Phrase clustering for discriminative learning. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, Suntec, Singapore, 2–7 August 2009; Association for Computational Linguistics: Stroudsburg, PA, USA, 2009; Volume 2, pp. 1030–1038.
27. Nothman, J.; Ringland, N.; Radford, W.; Murphy, T.; Curran, J.R. Learning multilingual named entity recognition from Wikipedia. *Artif. Intell.* **2013**, *194*, 151–175. [CrossRef]
28. Do Amaral, D.O.F.; Vieira, R. NERP-CRF: Uma ferramenta para o reconhecimento de entidades nomeadas por meio de Conditional Random Fields. *Linguamática* **2014**, *6*, 41–49.
29. McCallum, A.; Li, W. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2003; Volume 4, pp. 188–191.
30. Sears, T.D.; Sunehag, P. Induced semantics for undirected graphs: Another look at the Hammersley-Clifford theorem. In *AIP Conference Proceedings*; American Institute of Physics: College Park, MD, USA, 2007; pp. 125–132.
31. Jin, N. Ncsu-sas–Ning: Candidate generation and feature engineering for supervised lexical normalization. In Proceedings of the Workshop on Noisy User-Generated Text, Beijing, China, 31 July 2015; pp. 87–92.
32. Jiang, P.; Wu, H.; Wang, W.; Ma, W.; Sun, X.; Lu, Z. MiPred: Classification of real and pseudo microRNA precursors using random forest prediction model with combined features. *Nucleic Acids Res.* **2007**, *35* (Suppl. 2), W339–W344. [CrossRef] [PubMed]
33. Yadav, V.; Bethard, S. A survey on recent advances in named entity recognition from deep learning models. *arXiv* **2019**, arXiv:1910.11470.
34. Nie, Y.; Fan, Y. Arriving-on-time problem: Discrete algorithm that ensures convergence. *Transp. Res. Rec.* **2006**, *1964*, 193–200. [CrossRef]
35. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
36. Athiwaratkun, B.; Wilson, A.G.; Anandkumar, A. Probabilistic FastText for multi-sense word embeddings. *arXiv* **2018**, arXiv:1806.02901.
37. Ferreira, L.; Teixeira, A.; Cunha, J.P.S. REMMA-Reconhecimento de entidades mencionadas do MedAlert. In *Desafios na AvaliaçAo Conjunta do Reconhecimento de Entidades Mencionadas: O Segundo HAREM*; Linguateca: Aveiro, Portugal, 2008.

38. Pires, A.R.O. Named Entity Extraction from Portuguese Web Text. Master's Thesis, FEUP - Faculdade de Engenharia, Porto, Portugal, 2017.
39. Amaral, D.O.F.D. *Reconhecimento de Entidades Nomeadas na área da Geologia: Bacias Sedimentares Brasileiras*; PUCRS: Porto Alegre, Brazil, 2017.
40. Pirovani, J.P.C. CRF+ LG: Uma abordagem híbrida para o reconhecimento de entidades nomeadas em português. Ph.D. Thesis, Universidade Federal do Espírito Santo, Espírito Santo, Brazil, 2019.
41. Li, P.H.; Fu, T.J.; Ma, W.Y. Remedying BiLSTM-CNN Deficiency in Modeling Cross-Context for NER. *arXiv* **2019**, arXiv:1908.11046.