Journal of Intelligent & Fuzzy Systems 39 (2020) 2015–2025
DOI:10.3233/JIFS-179868
IOS Press

Probabilistic vs deep learning based approaches for narrow domain NER in Spanish

Orlando Ramos-Flores[a,*], David Pinto[a], Manuel Montes-y-Gómez[b] and Andrés Vázquez[a]
[a]*Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Puebla, México*
[b]*Coordinación de Ciencias Computacionales, Instituto Nacional de Astrofísica, Óptica y Electrónica, Santa María Tonantzintla, Puebla, México*

Abstract. This work presents an experimental study on the task of Named Entity Recognition (NER) for a narrow domain in Spanish language. This study considers two approaches commonly used in this kind of problem, namely, a Conditional Random Fields (CRF) model and Recurrent Neural Network (RNN). For the latter, we employed a bidirectional Long Short-Term Memory with ELMO's pre-trained word embeddings for Spanish. The comparison between the probabilistic model and the deep learning model was carried out in two collections, the Spanish dataset from CoNLL-2002 considering four classes under the IOB tagging schema, and a Mexican Spanish news dataset with seventeen classes under IOBES schema. The paper presents an analysis about the scalability, robustness, and common errors of both models. This analysis indicates in general that the BiLSTM-ELMo model is more suitable than the CRF model when there is "enough" training data, and also that it is more scalable, as its performance was not significantly affected in the incremental experiments (by adding one class at a time). On the other hand, results indicate that the CRF model is more adequate for scenarios having small training datasets and many classes.

Keywords: Named entity recognition, CRF, Bi-LSTM, Spanish, news reports

1. Introduction

The Named Entity Recognition (NER) task has been studied from the last two decades with the aim of extracting information from news, scientific documents, medicine records, social media, and other domains in different languages including Spanish. The term *named entity* was coined in the Message Understanding Conference-6 (MUC-6), where was introduced the task of recognizing names of people, organizations and geographical locations, as well as time, currency and percentage expressions in texts [11]. Several approaches have been proposed since then, for example, Maximum Entropy Markov Models (MEMM) [19, 27, 28] and Support Vector Machines (SVM) [2, 5, 14] were very popular in the first years. Later on, in the context of CoNLL-2002, [3] introduced a binary Adaptive Boosting (AdaBoost) algorithm to extract named entities in Spanish and Dutch, and several participants applied Conditional Random Fields (CRF) with great success. More recently, most works have considered Recurrent Neural Networks (RNN) approaches and have used pre-trained word embeddings to enhance the quality and generalization of data in the training process.

*Corresponding author. Orlando Ramos-Flores, Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Av. San Claudio y 14 Sur, C.P. 72570, Ciudad Universitaria, Puebla, México. E-mail: orlandxrf@gmail.com.

Although the extensive work in NER, it has been mainly addressed in English language, in the news domain, and in scenarios having large training datasets and a small number of classes to recognize. Most recent works have used probabilistic and deep learning approaches with competitive results, but there is not a clear and detailed comparison of them, which help to determine the best option to handle NER in Spanish under a narrow domain scenario consisting of a small and imbalanced dataset. This work focuses precisely on this problem.

Mainly, this work introduces an experimental study of two NER approaches, one probabilistic and one based on deep learning techniques, in two Spanish datasets, CoNLL-2002 (refer to Table 1) and MX-News (refer to Table 3). For the probabilistic approach, we used a Conditional Random Field model, whereas, for the deep learning approach, we applied two different Recurrent Neural Networks models, a Bidirectional Long-Short Term Memory, and a Bidirectional Long-Short Term Memory with ELMo embeddings. We selected these models because of their prominent results reported in the state-of-the-art.

The contribution of this paper is twofold. First, it compares the performance of two state-of-the-art NER approaches in two Spanish datasets and under incremental/decremental class scenarios. Second, it shows an in-deep analysis of the robustness, scalability and types of errors of each model, aiming at providing a detail characterization of them for their future application in Spanish related tasks. It is important to mention that the source code and datasets used in this paper are available in the GitHub repository[1].

The remainder of the paper is organized as follows. Section 2 describes some NER works using statistical and deep learning approaches. Then, Section 3 explains the used CRF and Bi-LSTM models. Section 4 reports the datasets used and experimental settings. Section 5 contains the experiments and results. Finally, the conclusions are presented in Section 6.

2. Related works

This section describes the two main current approaches for NER, one based on Conditional Random Fields (CRF approach from now on), and the other based on the use of recurrent and convolutional neural networks.

2.1. The CRF approach

This probabilistic model allows to segment and label data sequences [15]. It resolves the label bias problem, which usually affects the performance of Maximum-Entropy Markov Models (MEMMs), and it also shows greater robustness than Hidden Markov Models. In [25], it is proposed a CRF model to addressed the sequence labeling task, using shallow parsing features with POS tags and considering the IOB (Inside, Output, Begining) schema to label each word. Then, for the Fine-Grained Named Entity Recognition (FGNER) task, [17] used the CRF to detect the boundary of named entities and a Maximum Entropy (ME) model to classify them. The experiments considered fifteen base entities and one hundred forty-seven fine-grained categories. In other FGNER work, [24] proposed the distributed and parallelized feature extraction as well as the parameter estimation in CRF, to recognize six fine-grained geospatial concepts in German texts. On the other hand, the work of [22] reported a method that combines a logistic regression classifier and a CRF classifier to recognize food entities.

Moreover, the CRF approach has been used for recognizing medical entities in texts. For example, [1] proposed a hybrid method that combines semantic and statistical approaches under IOB schema to label entities from two medical corpora. [7] presented a feature generation method to incorporate multiple segmentation representations as IOB, IOBES, and BIES into a CRF model to achieve the NER task in biomedical and general domain corpora. The best results were obtained with BIES and IOBES. Similarly, [8] used eight different classifiers (CRF among them) to extend segmentation representations (IOB and IOBES included) on the medical i2b2-2010 corpus. In this work, an extra entity is used to represent entities ambiguity. The best results were obtained with the CRF classifier and the IOBES schema.

2.2. Neural networks approaches

Several types of neural networks architectures have been designed along the previous years, among them is the Bidirectional Long Short-Term Memory (Bi-LSTM), which has achieved the best results in the NER task. Bi-LSTMs are based in Recurrent Neural Networks (RNNs) [23] and Long Short-Term

[1] https://github.com/orlandxrf/spanish-ner

Memories (LSTMs) [12]; they were introduced in [9] and were compared against other networks architectures on the framewise phoneme classification task using the TIMIT corpus. The Bi-LSTM networks usually outperformed unidirectional LSTMs and are faster to train than RNNs. They have been extensively used in speech recognition because they are able to store past and future context internally. The experiments reported in [10] on two speech datasets showed state-of-the-art results.

Related to the NER task, [13] described several experiments in sequence tagging tasks, such as the Penn TreeBank POS tagging, the CoNLL-2000 chunking, and the CoNLL-2003 named entity tagging. An interesting contribution of this work was the combination of a Bi-LSTM with an extra CRF layer. They also considered spelling features, context features, word embeddings, and gazetteer features. In the experiments, they evaluated different combinations of LSTM networks, but the best results were achieved by the proposed Bi-LSTM-CRF network, even outperforming other state-of-the-art approaches. Following this work, [18] proposed a new neural network architecture for sequence labeling. Basically, it introduced an end-to-end model requiring no task-specific resources, feature engineering, nor data pre-processing; it only required word embeddings pre-trained on unlabeled data. The proposed model combined Convolutional Neural Networks (CNNs) [16] to encode character and word level representations that were used to feed a Bi-LSTM-CRF network. In the experiments, they used four different types of word embeddings, reporting better results than previous works. In the same direction, [6] implemented a Bi-LSTM-CNN using a character-level representation on the CNN layer and word embeddings (Senna[2], Glove[3], Word2vec[4]) on a Bi-LSTM network for the NER task. For the experiments in the CoNLL-2003 and OntoNotes 5.0 datasets, they considered a combination of word embeddings, capitalization features, and lexicon features.

Most works cited above have used large training datasets and have focused on recognizing a few types of named entities. Motivated by this fact, this work contributes in comparing state-of-the-art approaches, that is, Conditional Random Fields and Bidirectional Recurrent Neural Networks, when they are applied to more realistic scenarios, consisting narrow domain data, small training sets and many unbalanced classes[5].

[2]http://ronan.collobert.com/senna/

[3]https://nlp.stanford.edu/projects/glove/

[4]https://code.google.com/archive/p/word2vec/

3. Models used for the NER task

This section roughly explains the used CRF and Bi-LSTM models; their implementation details are described in Section 4.

3.1. CRF model

Conditional Random Fields are a sequence modeling framework introduced in [15]. They have all advantages of MEMMs, but solve the label bias problem. The main difference of the CRFs is that they use a single exponential model for the joint probability $p(y|x)$ of the entire sequence of labels given the observation sequence. In a CRF, two random variables are defined: $X = x_1, \ldots, x_T$, which is the variable over data sequences of observations (i.e., tokens) to be labeled, and $Y = y_1, \ldots, y_T$, which is the variable over the corresponding label sequences (i.e., named entity tags) [15, 26]. Formally, a linear-chain CRF can then be defined as follows:

$$p(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} exp\left(\sum_{k}^{K} \theta_k f_k(y_{t-1}, y_t, x_t)\right) \quad (1)$$

where f_k denotes one of K binary indicator (or feature) functions, each weighted by $\theta_k \in \mathbb{R}$, and Z is a normalization term, which iterates over all possible assignments.

$$Z(x) = \sum_{y} exp\left(\sum_{k}^{K} \theta_k f_k(y_{t-1}, y_t, x_t)\right) \quad (2)$$

The Figure 1 depicts an example of a CRF structure, where the sentence X is: "*López Obrador viaja a Puebla*" (*Lopez Obrador travel to Puebla*, in English), and the corresponding label sequence Y is: "*PER-B, PER-E, O, O, GPE-S*". The inputs and outputs are directly connected as opposed to LSTM and bidirectional LSTM networks, where memory cells/recurrent components are employed [13].

[5]See on https://github.com/orlandxrf/spanish-ner

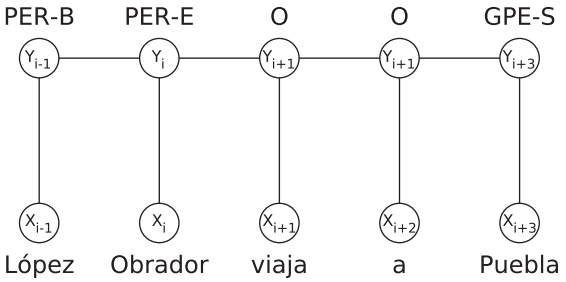

Fig. 1. Example of a CRF structure, where X_i correspond to the observation sequence and Y_i are the label sequence. Y_i tags are: **PER** for person, and **GPE** for geopolitical entity; they also include the labels **B** for begin, **E** for end, **S** for singleton, and **O** for output. Notation adapted from [13, 15].

3.2. Bidirectional long short-term memory

A Long Short-Term Memory (LSTM) is a special kind of RNN architecture proposed by Hochreiter and Schmidhuber [12] and [9]. It consists of a set of blocks recurrently connected, each one containing one or more recurrently connected memory cells, with the ability to remove or add information to the cell state. It is regulated by three multiplicative units: input, output and forget gates. Given an input sequence $x = (x_1, \ldots, x_T)$, a standard RNN computes the hidden vector sequence $h = (h_1, \ldots, h_T)$ and output vector of sequences $y = (y_1, \ldots, y_T)$ by iterating the following equations from $t = 1$ to T. In the NER task, x and y represent input features and tags respectively. In contrast, a LSTM contains three gates, which are functions of the current input x_t and hidden state h_t: input gate i_t, forget gate f_t and output gate O_t [10, 13]. The Figure 2 illustrates a single LSTM that is implemented as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (3)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (4)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (5)$$
$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (6)$$
$$h_t = o_t tanh(c_t) \quad (7)$$

where σ is the logistic sigmoid function, W terms denote weight matrices used for mapping the hidden layer input to three gates and the input cell state, b_i, b_f, b_o, b_c are bias vectors, and c is cell activation vectors, all of which are the same size as the hidden vector h [10, 13].

Figure 3 shows a labeling example using a bidirectional LSTM network over the same sentence depicted in Figure 1. The Bi-LSTM network allows accessing long-range context in both directions, that means that it can be trained using all available input

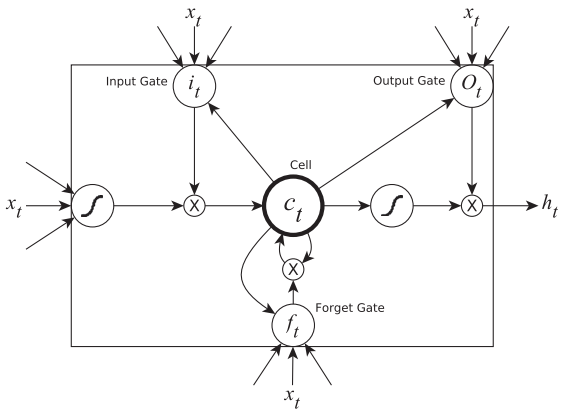

Fig. 2. Long Short-term Memory Cell. Source [10].

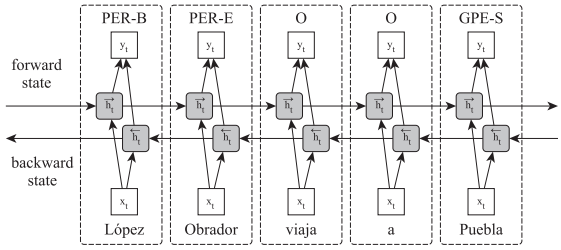

Fig. 3. A bidirectional LSTM network. Adapted from [13, 23].

information on the past (forward states) and future (backward states) of a specific time frame, unlike a LSTM network that only can use the previous context. In Figure 3, filled boxes represent the LSTM cells (they are also called the hidden LSTM layer). The bidirectional LSTM connects two hidden layers to a single output layer, both the hidden forward sequence ($\overrightarrow{h}_t$) and hidden backward sequence ($\overleftarrow{h}_t$) outputs are calculated using LSTM equations (3-7), iterating the forward layer from $t = 1$ to T and backward layer from $t = T$ to 1 [9, 10].

4. Experimental settings

This section presents the used datasets, the models' hyper-parameters as well as the evaluation measures.

4.1. Datasets

Two datasets were used in the experiments, the CoNLL-2002 Spanish corpus and MX-News corpus. The the CoNLL-2002 corpus contains four types of entities, organizations (ORG), persons (PER),

Table 1
CoNLL-2002 dataset. Sentences ∗indicate the original number of sentences in each dataset, maintaining their original length and used by the CRF model. Sentences †indicate the number of sentences used by the RNN model, which correspond to 50-tokens length sentences. Tokens indicate the size of the vocabulary in each partition

	Test-A	Test-B	Train	Ensemble
Sentences ∗	1,915	1,517	8,323	11,755
Sentences †	2,177	1,848	9,947	13,972
Tokens	9,646	9,086	26,099	31,405
Individual tags	8	8	8	8
Schema	IOB (Inside/Output/Begining)			

Table 2
Tags used for the annotation of entities in the MX-News corpus

No.	Tag	Description
1	PER	People names, aliases and abbreviations.
2	ORG	Organizations, institutions.
3	DAT	Dates on different formats.
4	TIT	Title or position of persons.
5	GPE	Country names, states, cities, municipalities.
6	PEX	Political party names, aliases and abbreviations.
7	TIM	Time expresions.
8	FAC	Facility names.
9	EVT	Event names.
10	ADD	Addresses expressions, URLs and Twitter users.
11	MNY	Monetary amounts.
12	DOC	Documents, laws, rules.
13	PRO	Product names, brands, application names.
14	PRC	Percentage expressions.
15	DEM	Geographical or racial origin of people.
16	AGE	People age.
17	LOC	Locations about regions, rivers, lakes.

locations (LOC), and miscellaneous (MISC). This corpus was tagging under the IOB schema, and it contains three partitions, TestA + TesB + Train. In our experiments we considered all these partitions as well as their union (refered as *Ensemble*). Table 1 shows some statistics from this dataset. It is important to clarify that for the RNN model the sentences longer than 50 tokens were divided into small sentences, and sentences shorter than 50 tokens were padded with the special "<*pad*>" token.

The MX-News dataset was gathered by ourselves. It consists of 250 political news documents from Mexico, which were manually labeled with seventeen different types of entities as described in Table 2. The labeling is done based on the IOBES tagging schema [28, 14], as it has shown better results than the traditional IOB schema [7, 8]. The boundaries of the named entities (NEs) are marked with the **B** and **E** tags. The tags **I** and **S** are used to represent tokens inside the NE and NEs of one single word respectively.

Table 3
MX-News dataset. Sentences ∗indicate the original number of sentences in each split, maintaining their original length and used by the CRF model. Sentences †indicate the number of sentences used by the RNN model, which correspond to 50-tokens length sentences. Tokens indicate the size of the vocabulary in each slit

	Split I	Split II	Split III	Ensemble
Sentences ∗	1,295	1,295	1,297	3,888
Sentences †	1,666	1,677	1,661	5,004
Tokens	7,628	7,726	7,664	13,273
Individual tags	63	63	63	65
Schema	IOBES (Inside/Output/Begining/End/Single)			

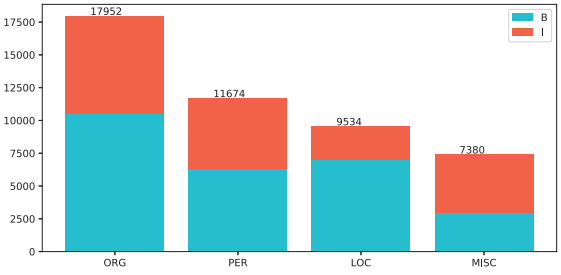

Fig. 4. CoNLL-2002 Corpus: Class distribution under the IOB schema.

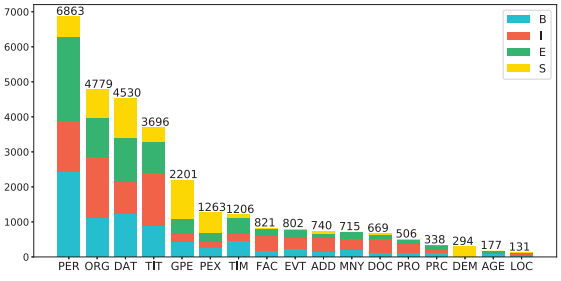

Fig. 5. Mexican News Corpus (MX-News): Class distribution under the IOBES schema.

To summarize, Figure 4 shows the classes distribution of the CoNLL-2002 Spanish corpus under IOB tagging schema. Similarly, Figure 5 shows the classes distribution of the MX-News corpus using the IOBES tagging schema. As noticed, both datasets present high class imbalance, particularly the MX-News corpus.

4.2. Implementation details of the CRF model

The features used in the CRF model were: word suffixes, simplified POS tags, flags indicating the use of lowercase, uppercase and digits, marks for titles as well as for the begin and end the sentences, and features of nearby words. The implementation of this

model was done using the CRFsuite library[6], with the following hyperparameters:

- *algorithm* = *lbfgs* (Gradient descent using the L-BFGS method).
- *c1* = 0.1 (coefficient for L1 regularization).
- *c2* = 0.1 (coefficient for L2 regularization).
- *max_iterations* = 50 (The maximum number of iterations for optimization algorithms).
- *all_possible_transitions* = *True* (CRFsuite generates transition features that associate all possible label pairs).

4.3. Implementation details of the Bi-LSTM model

To implement this model we used Keras 2.2.4[7], and TensorFlow 1.13.1 as backend. The implemented model consisted of 4 layers:

- *Input Layer*. The input sentences to the model. The length of sentences was defined to be 50 words. Shorter sentences were filled with the "<pad>" word and longer sentences were split into 50 words instances.
- *Embedding Layer*. Each word within the vocabulary was mapped to a vector of 1024 size using the ELMo embeddings [21]. The parameters from this layer were: *input_dim* is the vocabulary size, *output_dim* = 1024, *input_length* = 50, *weights* = embeddings matrix ELMo and *trainable* = *True*.
- *Bidirectional LSTM Layer*. This layer was feed by the embedding layer. The LSTM layer was set up with *units*=200, *dropout* = 0.01, *recurrent_dropout* = 0.3 and *return_sequences* = *True*. The activation function was the Hyperbolic tangent (tanh) and for the recurrent step it was the hard sigmoid function. A Bidirectional wrapper layer was used to learn high-level features in the forward and backward directions of the LSTM layer.
- *Output Layer*. *Time Distributed* is a wrapper layer applying the same dense layer (same weights) to the LSTMs outputs for one time step at a time. In this way, the output layer only needs one connection to each LSTM unit (plus one bias). It uses the parameters: *units* = positive integer, the dimensionality of the output space (*tags length*) and *activation* = *softmax* function.

For training the model we employed the following configuration: the RMSprop algorithm as optimizer, with *learning rate* of 0.001 and *learning rate decay* of 0.0. For the evaluation stage we considered the following parameters: *metrics* = *accuracy* and *loss* = *categorical_crossentropy*. The *bach_size* = 50, *epochs* = 20, *validation_split* = 0.2 and *shuffle* = *True*.

In this work, we used ELMo embeddings [21] in Spanish. The embeddings were built from both corpora (CoNLL-2002 and MX-News), using the *elmoformanylangs 0.0.2* Python Library based on [4], which built pre-trained ELMo representations for many languages. The embeddings length was 1024 for each token.

4.4. Evaluation

The comparison of the NER approaches was done based on two types of evaluation. The first type considered the individual tags for each word from the NEs[8], in both tagging schemes, IOB and IOBES. Therefore, in the CoNLL-2002 corpus that has 4 NEs and 2 tags under the IOB schema (I and B), 8 (4x2) individual tags were evaluated. In contrast, in the MX-News corpus that has 17 NEs and 4 tags under the IOBES schema (I, B, E, and S), 65 individual tags were evaluated. We could not evaluate the 68 (17x4) tags because the corpus does not contain examples of all of them as shown in Figure 5. It is important to clarify that in both schemes, the tokens labeled with the letter "O" indicate that they are not named entities.

The second evaluation type considered the complete NEs, without taking into account the information from the IOB or IOBES schemes. That is, in this type of evaluation a named entity labeled as PER-B PER-I PER-E will be only taken as PER PER PER. In other words, a named entity will be considered as correctly recognized if all their individual tags correspond to the same class (PER in our example).

For the two types of evaluation we used the same measures, the macro-average of precision, recall, and F1[9]. To be more precise, we first computed each measure for each one of the NEs and then we computed the average over all their types, as suggested in [20].

[6] http://www.chokkan.org/software/crfsuite/

[7] https://keras.io/

[8] For example, the name of a person such as "*Andrés Manuel López Obrador*" has the tags PER-B, PER-I, PER-I, PER-I in the IOB schema.

[9] For the first evaluation type we used the measures from the *scikit-learn* [10] *0.20.3* Python library metrics, whereas for the second type of evaluation we used the *seqeval* [11] *0.0.10* Python library.

Table 4
Results from the three models in the CoNLL-2002 (Cn) and MX-News (Mx) datasets. Best results are marked in bold

Model	D	P_i	R_i	$F1_i$	P_c	R_c	$F1_c$
CRF	Cn	0.83	0.80	0.81	0.84	0.82	0.83
	Mx	0.86	0.74	0.78	**0.93**	**0.88**	**0.90**
BL	Cn	0.78	0.75	0.76	0.76	0.76	0.76
	Mx	0.77	0.71	0.73	0.81	0.83	0.82
BLE	Cn	**0.87**	**0.86**	**0.86**	0.83	0.87	0.85
	Mx	0.85	0.75	0.78	0.84	**0.88**	0.86

Furthermore, for the evaluation of individual tags, macro-average F1 is labeled as $F1_i$, while for the evaluation of complete NEs, the macro-average F1 is labeled as $F1_c$.

5. Experimental results

The main purpose of the experiments was to analyze the performance of the CRF, Bi-LSTM (BL), and Bi-LSTM-ELMo (BLE) models in the CoNLL-2002 and MX-News datasets.

5.1. General comparison of the models

This first experiment compares the general performance of the probabilistic and deep learning models in the two datasets, using their *ensemble* partitions which include all available data. Results from the three models are shown in Table 4. It shows the precision, recall and F1 for the two kinds of evaluation types, where index i is used to indicate the individual tags evaluation and index c is employed for the complete-NE evaluation.

Table 4 shows that in general the complete-NE results outperform those from the individual-tags evaluation, because of the finer granularity of the latter. Furthermore, these results indicate that the Bi-LSTM-ELMo was the best model for the CoNLL-2002 dataset, regardless of the type of evaluation, whereas the CRF model obtained the best results for the MX-News dataset. These results allow us to formulate the following initial conclusions: for scenarios having large amounts of data and considering few NE classes, the Bi-LSTM-ELMo model is a suitable option, but for the opposite case, consisting of small training sets and many classes, the CRF is a better selection.

5.2. Performance under similar conditions

For this experiment, the MX-News dataset was reduced from seventeen to four classes (refer hereon as Mex-News-4), and its annotation schema was changed from IOBES to IOB. Thus, both datasets had the same number of clasess and used the same tagging schema. The Mex-News-4 dataset preserved the PER, ORG and GPE[12] classes, while the rest were integrated to the MISC class. Figure 6 shows the tags distributions in both datasets; for all classes except the MISC class the CoNLL-2002 has more data than the MX-News-4 corpus, however, for all classes both datasets show similar distributions of B and I labels.

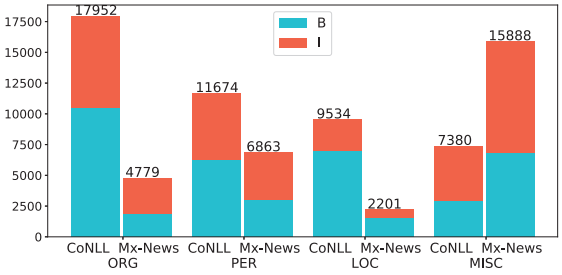

Fig. 6. Tags distributions in the CoNLL-2002 and MX-News-4 datasets.

Table 5 shows the results from this experiment, which allow to compare the probabilistic and deep learning models in two collections sharing most features except the quantity of available training data. The obtained results confirm our previous observations, that is, the Bi-LSTM-ELMo is more effective than the other approaches when there is "enough" training data (e.g., it obtained the best results in the CoNLL corpus) and, on the other hand, the CRF model is very competitive, especially for scenarios having limited training data (as the case of the MX-News-4 dataset). Once again the Bi-LSTM (BL) model showed the worst performance; we presume this is because this network does not use external word embeddings and, in consequence, it has difficulties to extract general discriminative patterns.

Results from Table 5 also show that in general the complete-NE results outperform those from the individual-tags evaluation. They also show that $F1_c$ scores of the three models are higher for the MX-News-4 corpus than for the CoNLL-2002 dataset, which could be explained by the highest level of narrowness of the MX-News dataset, i.e., both datasets belong to the news domain, but the MX-News datasets exclusively contains politics news.

[12] It was renamed as LOC, to use the same labels as the CoNLL corpus.

Table 5
Results from the three models in the CoNLL-2002 (Cn) and MX-News-4 (Mx) datasets

Model	D	P_i	R_i	$F1_i$	P_c	R_c	$F1_c$
CRF	Cn	0.83	0.80	0.81	0.84	0.82	0.83
	Mx	**0.91**	0.88	0.89	**0.90**	0.85	**0.87**
BL	Cn	0.78	0.75	0.76	0.76	0.76	0.76
	Mx	0.86	0.85	0.85	0.79	0.82	0.81
BLE	Cn	0.87	0.86	0.86	0.83	0.87	0.85
	Mx	0.90	**0.90**	**0.90**	0.84	**0.88**	0.86

Table 6
F1 scores in the different partitions of the CoNLL-2002 and MX-News-4 datasets

CoNLL-2002				MX-News-4			
1	2	3	4	1	2	3	4
0.76	0.73	0.77	0.81	0.86	0.84	0.87	0.89
0.76	0.78	0.78	0.83	0.83	0.80	0.84	0.87
0.67	0.66	0.73	0.76	0.81	0.80	0.80	0.85
0.64	0.66	0.70	0.76	0.74	0.68	0.70	0.81
0.81	0.82	0.83	**0.86**	0.88	0.84	0.87	**0.90**
0.81	0.81	0.83	0.85	0.82	0.77	0.82	0.86

5.3. Robustness of the models

These experiments have the purpose of analyzing the robustness and scalability of the three models. To do that, we carried out two kinds of experiments; the first one evaluated the performance of the models in the different partitions of the two datasets, the second experiment used several data subsets consisting of different number of classes and distributions. This second experiment was performed only in the MX-News dataset.

Table 6 shows the F1 results corresponding to the first experiment. Horizontal lines divide the results from the three models, namely, CRF, Bi-LSTM, and Bi-LSTM-ELMo. Inside each division, the first rows of results correspond to the complete-NE evaluation (c), whereas the second rows represent the individual tags evaluation (i). The results indicate that the Bi-LSTM-ELMo model is the most robust to the changes in the dataset, since its results show the lowest variations. In addition, all the models tend to be more stable or robust in the complete-NE evaluation than in the individual tags evaluation.

For the second experiment, we applied some incremental/detrimental procedures to generate different evaluation data subsets. Figure 7 illustrates these procedures, where C_i indicates the class i, and classes are ordered and integrated from sparse to dense and vice versa. In the ascendant procedure classes are aggregated from sparse to dense, once at a time. In contrast,

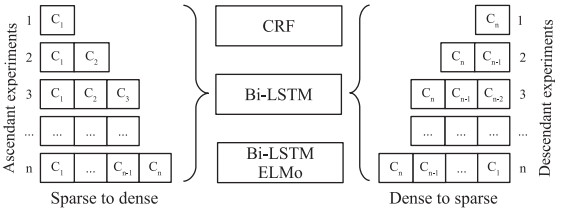

Fig. 7. Generation of data subsets for evaluating the robustness of models in the Spanish NER task.

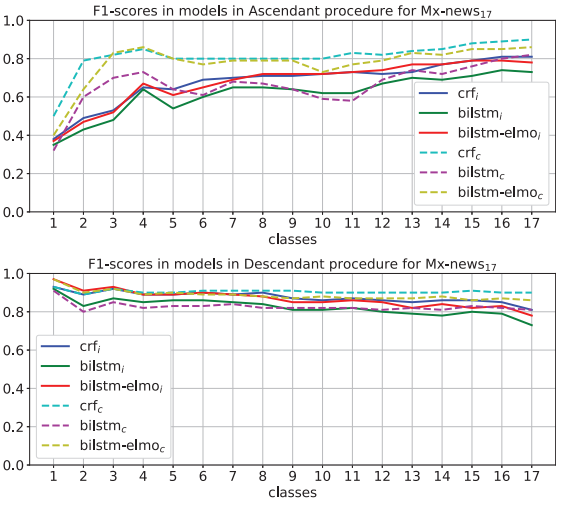

Fig. 8. $F1_i$ scores of the three models in the different data subsets from the MX-News corpus.

in the descendant procedure classes are included from dense to sparse. The experiments considered the application of three models (CRF, Bi-LSTM, and Bi-LSTM-ELMo) over the ensemble partition of the MX-News dataset.

The plots from Figure 8 show the performance of the three models in the incremental experimental setting, which consists in aggregating one class at a time starting from the sparse to the dense classes (plot on the top) and vice versa (plot on the bottom). These plots include results from both types of evaluations, the individual tags evaluations are represented by continuous lines, while the complete-NE evaluations are indicated by dotted lines. The numbers in the x-axis correspond to the number of classes considered at each experiment. Results from these plots are very interesting, they indicate that the models are more sensitive, or less robust, to the number of examples from each class than to the number of classes. This is evident in the first part of the plots, where the results from the descendant procedure (which first considers dense classes) are much more better than the results

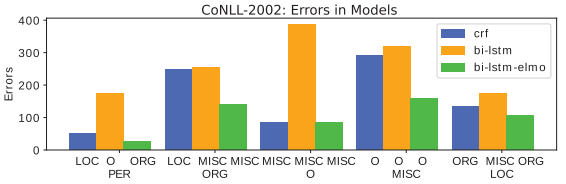

Fig. 9. Errors in Models on CoNLL-2002 dataset.

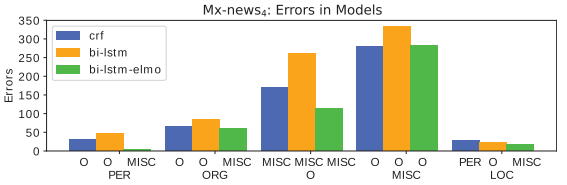

Fig. 10. Errors in Models on MX-News-4 dataset.

from the ascendant procedure (which first considers sparse classes). Moreover, in the descendant procedure, aggregating more classes (moving from 1 to 17 classes) did not cause a significant lost in effectiveness. It is also important to notice that the CRF model obtained the best results in both incremental procedures, which confirms that it is the more robust model for NER in scenarios with many classes and with few examples per class.

5.4. Error analysis

This section focuses on the analysis of the number and kind of errors generated by the three models, with the aim of determining how complementary or redundant they are. Figures 9 and 10 describe the main kind of errors generated by the three models in the CoNLL-2002 and MX-News-4 collections. In the bottom part of these figures it is indicated the correct classes of the entities, and just above them the most frequent erroneous classes predicted by each of the models. The height of the bars were determined from the information of the confusion matrices; the class "O" was included to show errors related to labeling a named entity as non-entity or a common word as a named entity.

Figures 9 and 10 show some interesting patterns. First, the errors of the three models were slightly different in the two datasets, which means that they are mainly explained by the nature of the data, more than supported by models themselves. For example, LOC entities were usually confused with ORG entities in the CoNLL-2002 corpus, but they were confused with PER and MISC entities in the MX-News-4 dataset. Second, the three models show different errors in general, but they share two of them: they tended to classify words not belonging to any named entity (labeled as O) as miscellaneous entities (MISC) and vice versa.

Finally, Tables 7, 8 and 9 list the most frequent mistakes, generated by any of the models, in the CoNLL-2002, MX-News-4 and MX-News-17 datasets respectively. In the heading of the tables, **C** is the correct class of the named entity, and **NE** the name of the incorrectly-recognized entity. For both headings, the subindex refers to the type of evaluation, *i* indicates the individual tags evaluation and *c* the complete-NE evaluation. The numbers besides the name entities indicate the frequency of occurrence of the entities. From these tables it is possible to observe that the most common errors were different for the two types of evaluation, and for the two datasets.

Table 7
Top mistakenly NE scores recognized for CoNLL-2002

Ci	NEi_f	Cc	NEc_f
LOC	Madrid$_{54}$	ORG	EFE$_{66}$
LOC	Barcelona$_{46}$	ORG	China$_{54}$
LOC	España$_{34}$	ORG	Estado$_{44}$
LOC	Europa$_{28}$	ORG	ESP$_{40}$
LOC	Cuba$_{28}$	ORG	Rusia$_{31}$
LOC	Zaragoza$_{23}$	LOC	España$_{30}$
LOC	Chile$_{22}$	ORG	España$_{27}$
LOC	Argentina$_{21}$	ORG	FRA$_{24}$

Table 8
Top mistakenly NE scores recognized for MX-News-4

Ci	NEi_f	Cc	NEc_f
LOC	México$_{42}$	MISC	Morena$_{37}$
LOC	EU$_{30}$	MISC	mexicano$_{35}$
LOC	Oaxaca$_{22}$	MISC	2018$_{33}$
LOC	Estado$_{18}$	MISC	Elección 2018$_{32}$
LOC	S.L.P$_{16}$	MISC	presidente$_{26}$
LOC	San Luis Potosí$_{16}$	MISC	1 de julio$_{24}$
ORG	Nacional$_{15}$	MISC	gobernador$_{17}$
LOC	Estados Unidos$_{13}$	MISC	alcalde$_{17}$

Table 9
Top mistakenly NE scores recognized for MX-News-17

Ci	NEi_f	Cc	NEc_f
DEM	mexicano$_{72}$	TIT	presidente$_{157}$
PEX	Morena$_{59}$	ORG	Gobierno$_{143}$
TIT	presidente$_{40}$	GPE	México$_{91}$
AGE	años$_{37}$	PEX	Morena$_{90}$
TIM	mañana$_{32}$	TIT	virtual presidente de México$_{74}$
DAT	2018$_{32}$	ORG	República$_{68}$

6. Conclusions

This work presented an experimental study in the task of Named Entity Recognition. This study aimed at comparing probabilistic and deep learning models in recognizing named entities in a narrow domain scenario in Spanish language. For the experiments three different models were considered, one Conditional Random Field model and two Bidirectional Long-Short Term Memories, and two different collections were used, the CoNLL-2002 corpus with four classes labeled under IOB tagging schema, and the MX-News corpus with seventeen classes labeled under IOBES schema.

The obtained results allowed us to formulating the following conclusions: (i) the Bi-LSTM-ELMo model is a better option than the CRF model for scenarios having small number of classes and a big number of training examples, on the contrary, the CRF model result to be a better option for the opposite cases; (ii) both kind of models, probabilistic and deep learning based, were robust to the variations of the datasets when the evaluation was done at NE level, but were not equally stable when considering the evaluation at word level (i.e., individual tags); (iii) the number of training instances per class influences or affects more the effectiveness of the models than the number of classes (types of NEs to be recognized), being the CRF model the most robust to difficult conditions as well as the most scalable to consider large number of classes; (iv) the number and kind of errors from the three models are considerably different, and therefore they tend to be complementary to each other, especially the CRF and the Bi-LSTM-ELMo models.

As future work, we plan to extend this study by considering other neural networks models like CNNs and hybrid models (CNN + Bi-LSTM + CRF). We also plan to consider fine-grained named entities in other languages and domains, with aim to verify the performance of models in different scenarios.

References

[1] A.B. Abacha and P. Zweigenbaum, Medical entity recognition: A comparison of semantic and statistical methods, In *Proceedings of BioNLP 2011 Workshop*, BioNLP '11, pages 56–64, (2011). ISBN 978-1-932432-91-6.

[2] M. Asahara and Y. Matsumoto, Japanese named entity extraction with redundant morphological analysis, In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, volume 1 of *NAACL '03*, pages 8–15, (2003).

[3] X. Carreras, L. Màrquez and L. Padró, Named entity extraction using adaboost, In *Proceedings of the 6th Conference on Natural Language Learning*, volume 20 of COLING-02, pages 1–4, Stroudsburg, PA, USA, (2002). Association for Computational Linguistics.

[4] W. Che, Y. Liu, Y. Wang, B. Zheng and T. Liu, Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation, In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium, October (2018). Association for Computational Linguistics.

[5] S. Chesney, G. Jacquet, R. Steinberger and J. Piskorski, Multiword entity classification in a highly multilingual environment, In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 11–20, (2017).

[6] J.P. Chiu and E. Nichols, Named entity recognition with bidirectional lstm-cnns, *Transactions of the Association for Computational Linguistics* **4** (2016), 357–370.

[7] H.-C. Cho, N. Okazaki, M. Miwa and J. Tsujii, Named entity recognition with multiple segment representations, *Information Processing & Management* **49**(4) (2013), 954–965. ISSN 0306-4573.

[8] A. Goyal, V. Gupta and M. Kumar, Recent named entity recognition and classification techniques: A systematic review, *Computer Science Review* **29** (2018), 21–43. ISSN 1574-0137.

[9] A. Graves and J. Schmidhuber, Framewise phoneme classification with bidirectional lstm and other neural network architectures, *Neural Networks* **18**(5) (2005), 602–610. ISSN 0893-6080. IJCNN 2005.

[10] A. Graves, N. Jaitly and A. Mohamed, Hybrid speech recognition with deep bidirectional lstm, In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 273–278, (2013). doi:10.1109/ASRU.2013.6707742

[11] R. Grishman and B. Sundheim, Message understanding conference-6: A brief history, In *Proceedings of the 16th Conference on Computational Linguistics, volume 1 of COLING '96*, pages 466–471, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics.

[12] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation* **9**(8) (1997), 1735–1780. doi:10.1162/neco.1997.9.8.1735

[13] Z. Huang, W. Xu and K. Yu, Bidirectional lstm-crf models for sequence tagging, *CoRR*, abs/1508.01991, (2015).

[14] T. Kudo and Y. Matsumoto, Chunking with support vector machines, In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, NAACL '01, pages 1–8, (2001).

[15] J.D. Lafferty, A. McCallum and F.C.N. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-778-1.

[16] Y. Le Cun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard and L.D. Jackel, Backpropagation applied to handwritten zip code recognition, *Neural Computation* **1**(4) (1989), 541–551.

[17] C. Lee, Y.-G. Hwang, H.-J. Oh, S. Lim, J. Heo, C.-H. Lee, H.-J. Kim, J.-H. Wang and M.-G. Jang, Fine-grained named entity recognition using conditional random fields

for question answering, In H.T. Ng, M.-K. Leong, M.-Y. Kan and D. Ji, editors, *Information Retrieval Technology*, pages 581–587, Berlin, Heidelberg, (2006). Springer Berlin Heidelberg. ISBN 978-3-540-46237-8.

[18] X. Ma and E.H. Hovy, End-to-end sequence labeling via bidirectional lstm-cnns-crf, *CoRR*, abs/1603.01354, 2016.

[19] A. McCallum, D. Freitag and F.C.N. Pereira, Maximum entropy markov models for information extraction and segmentation, In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 591–598, (2000). ISBN 1-55860-707-2.

[20] A. Özgür, L. Özgür and T. Güngör, Text categorization with class-based and corpus-based keyword selection, In *Computer and Information Sciences - ISCIS 2005*, pages 606–615, Berlin, Heidelberg, (2005). Springer Berlin Heidelberg. ISBN 978-3-540-32085-2.

[21] M.E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee and L. Zettlemoyer, Deep contextualized word representations, In *Proceedings of NAACL-HLT*, (2018).

[22] T. Sasada, S. Mori, T. Kawahara and Y. Yamakata, Named entity recognizer trainable from partially annotated data, In K. Hasida and A. Purwarianti, editors, *Computational Linguistics*, pages 148–160, Singapore, (2016). Springer Singapore. ISBN 978-981-10-0515-2.

[23] M. Schuster and K.K. Paliwal, Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing* **45**(11) (1997), 2673–2681. ISSN 1053-587X. doi:10.1109/78.650093

[24] R. Schwarzenberg, L. Hennig and H. Hemsen, In-memory distributed training of linear-chain conditional random fields with an application to fine-grained named entity recognition, In G. Rehm and T. Declerck, editors, *Language Technologies for the Challenges of the Digital Age*, pages 155–167, Cham, (2018). Springer International Publishing. ISBN 978-3-319-73706-5.

[25] F. Sha and F. Pereira, Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL'03, pages 134–141, (2003).

[26] C. Sutton and A. McCallum, An introduction to conditional random fields, *Foundations and Trends®in Machine Learning* **4**(4) (2012), 267–373.

[27] B.T. Todorović, S.R. Rančić and E.H. Mulalić, Context Hidden Markov Model for Named Entity Recognition, pages 447–460. Springer New York, New York, NY, (2011). ISBN 978-1-4419-6594-3.

[28] K. Uchimoto, Q. Ma, M. Murata, H. Ozaku and H. Isahara, Named entity extraction based on a maximum entropy model and transformation rules, In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL'00, pages 326–335, (2000).