Neural-based approach for Cancer Named Entity Recognition task in Spanish

José Ignacio Baciero Fernández[a,b]

Abstract

This document presents both the working notes, neural-based model and conclusions regarding the development of a Named Entity Recognition model to achieve identification of mentions of cancer injuries and its related symptoms in Spanish written medical records.

The recognition of any kind of concept in natural text (called entity recognition, *NER*) is a core technology for automatic or semi-automatic analysis of textual resources. *NER* task is the previous step for many applications that work on text, such as information retrieval, information extraction, or document classification. In recent years, the problem has received considerable attention in the *bioinformatics community* and experience has shown that *NER* in the life sciences is a rather difficult problem. Several systems and algorithms have been developed and applied in the field of biomedical sciences. This paper describes how this task is performed in some Spanish medical records, and how the algorithm has been developed the application of *NER* research.

Keywords

Natural Language Processing, Named Entity Recognition, CNN, spaCy, Deep Learning, Python, Transfer Learning, CANTEMIST 2020

1. Introduction

Clinical records are the key resource to help study strategies to identify and prevent a wide range of medical disorders. However, due to the exponential increase in health dossiers, it has become a very costly and time consuming task to manually review this amount of information. Therefore, delivering systems able to systematically identify and classify key entities or phrases of interest in those records is a vital task.

Before the discovery of the Machine Learning models and the improvement of Neural Networks based techniques, the analysis in the Natural Language field was made attending to morphological, syntactic and lexical rules previously parameterized. With these approaches, satisfactory results were obtained but they have been surpassed to a great extent by the entry into the scene of the mentioned Networks.

IberLEF 2020: Iberian Languages Evaluation Forum, Tuesday, 22 September 2020

bacierofdez.5@gmail.com (J. I. B. Fernández)
https://github.com/IBacieroFdez/ (J. I. B. Fernández)
0000-0001-6791-354X (J. I. B. Fernández)

CEUR Workshop Proceedings (CEUR-WS.org)

1.1. State of Art of Named Entity Recognition

Models with very high yields have already been obtained in languages such as English. However, in other languages, including Spanish, very poor performances are obtained in very specific domains. With this project, the aim is to achieve a first approach in this field in the Spanish language.

NER is one of the most promising fields in biomedicine. It has been studied for approximately thirty years. In this time, these systems have grown considerably in complexity, from simple rule-based pattern comparators to sophisticated hybrid machine learning classifiers.

It is very plausible that this growth in sophistication has also resulted in improved efficiency. It is unlikely that it will be possible to improve the NER problem much further in the general classes, but there is likely to be progress in the specialized areas. In particular, species-specific NER is a promising direction, but currently remains constrained by the lack of sufficiently large and species-specific corpus. If at least some samples can be labeled by an automated system with very high precision, NER systems can consider surrounding words to learn reliable context patterns.

At this point of the play comes the role of this Shared Task, conceived to address this issue in the study of cancer and its symptoms. Cantemist first shared task consists in the train and development of a NER model with more than a thousand labeled samples of anonymized medical records.

1.2. Data Set description

Throughout the following working notes a tour through each and every one of the phases that make up the work will be carried out, from the phase of adapting the texts to training data, working with the "train-set-to-publish", "dev-set1-to-publish" and "dev-set2-to-publish" data sets to perform the training of this model to the phase of evaluation of the NER tool. The last phase will be carried out with the texts belonging to the dataset named "test-background-set-to-publish".

As a result of the completion of all the so mentioned steps, this model by itself will be able to carry out the identification of any word or group of words as belonging to a semantic category; in this case, cancer related and represented by the name of "**MORFOLOGIA_NEOPLASIA**". With this we can distinguish some words from others according to the meaning they offer in the context in which they are found.

2. Algorithm Description

It is considered beneficial to start from a *pre-trained DL* model because Transfer Learning provides you with a first approximation to the information you will be trained with. That is, it allows you to calculate the first weights and biases before the actual training.

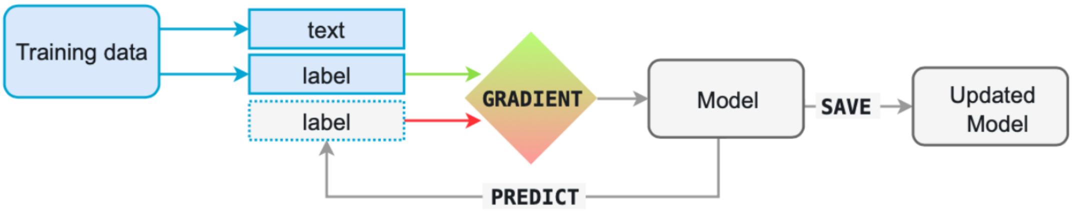

Figure 1: Training flowchart of a base model of spaCy, Source: SpaCy statistical model training (https://spacy.io/usage/training).

These weights and biases will be nowhere near what the model will have after training, but it avoids taking random weights and biases, with the cost of computer time and examples of the training data set invested in this task.

As previously specified, the training data will be represented by one Nominal entity type. This nominal entity type is the already presented "MORFOLOGIA_NEOPLASIA".

2.1. spaCy CNN Pre-trained model

The model to be used from ***spaCy*** Python library is called **"es_core_news_md"** and consists on Spanish multi-task *CNN* trained on *UD Spanish AnCora* and *WikiNER*. Assigns word vectors, POS tags, dependency parses and named entities. Word vectors trained using FastText CBOW on Wikipedia and OSCAR (Common Crawl).

SpaCy models have been designed and implemented specifically to give the user a balance in speed, size and accuracy properties. They use convolutional layers with residual connections, layer normalisation and maximum non-linearity, giving much better efficiency than the standard BiLSTM solution.

The parser and NER use a mimicry learning objective to provide accuracy in line with the latest research systems, even when evaluating from raw text allowing them to be 10 times smaller, 20 % more accurate, and even cheaper to run than the previous generation.

2.2. Fine tuning of model hyperparameters

Not only the amount of Nominal Entities to "learn" and the amount of data set partitions will distinguish the different models; also certain hyper-parameters of the data set will be varied in order to find the ideal combination of these at the same amount of training and evaluation data. These parameters are the number of iterations per training session and the forgetting rate.

The greater the number of iterations the better the accuracy with the possible unfavourable effect of memorising the particular examples and the consequent inability to generalise. Conversely, the forgetting rate represents the percentage of connections in the neural network that are "forgotten" (removed) during training to avoid this memorisation of training data.

The fact that a model memorizes the data with which it is trained is not a very accepted quality in the community because even if the model obtains good performances in the domain in which it is trained (with the training data), it will obtain bad results in the evaluation against texts belonging to other domains. This effect is known as Overfitting.

On the other hand, it is also not good to abuse a high dropout rate and a small number of iterations during the training session, as this could result in insufficient training, known as Underfitting.

Bearing this in mind, the following values for Drop Out rate and iterations per training epoch have been selected:

Table 1
Hyperparameter Values

Drop Out rate	Iterations per epoch
0.15	50

10.5281/zenodo.4000589

3. Results of the evaluation stage and Conclusions

NLP systems are typically evaluated with respect to their performance in the task-specific test suite.

- Accuracy: measures the system's ability to correctly identify and classify the entities present in the text.
  - This equation penalizes the algorithm that estimates that pieces of text are entities that really are not or misclassifies those entities. The formula:

$$Acc = \frac{T_P}{T_P + F_P} \tag{1}$$

- Recall (or Completeness): this metric measures the ability of the algorithm to detect the entities that are present in the text.
  - That is, this function penalizes algorithms that overlook and do not estimate as chunky entities in the text that they really are. The formula:

$$Rec = \frac{T_P}{T_P + F_N} \tag{2}$$

- Formula F1: is a metric that is used to combine the results of the two previous formulas into a single number. Furthermore, for the classification of multiple classes, the score F1 is used, which is the harmonic mean of Precision and Recall.
  - To avoid confusion, this formula generates an average between the previous results; thus providing a good estimate of the overall quality of a model. The formula:

$$F1 = 2\frac{P \cdot R}{P + R} \tag{3}$$

where T_P, T_N, F_P and F_N are the number of true positives, true negatives, false positives and false negatives respectively. Intuitively, the number of true predictions is divided by the number of all predictions.

3.1. Statistical NER model results

As described in the CANTEMIST evaluation guidelines, the performance of the *NER* model is defined by the previously mentioned F1 metric. The table below presents Accuracy and Recall as well as F1:

Table 2
NER Model Performance

Accuracy	Recall	F1 Metric
0.808	0.802	0.805

4. Conclusions

Throughout this report, the training and evaluation tasks of the ***"baciero-fdez"*** name system involved in subtask 1 proposed by CANTEMIST-2020 have been described. It takes advantage of the technology offered by spaCy, a tool based on deep learning with bi-directional layers of LSTM and CRF for the NER task. Considering the challenges described above, our system reaches a metrica-F1 of 80.5%, a competitive and satisfactory result so far.

Acknowledgments

Thanks to the developers of ACM consolidated LaTeX styles https://github.com/borisveytsman/acmart and to the developers of Elsevier updated LaTeX templates https://www.ctan.org/tex-archive/macros/latex/contrib/els-cas-templates.

10.5281/zenodo.4000589

5. Bibliography

(1) LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324.

(2) Bachani, N. (s. f.). Chunking in NLP: decoded. April 1st, 2020, https://towardsdatascience.com/chunking-in-nlp-decoded-b4a71b2b4e24

(3) Francois Chaubard, Rohit Mundra, Richard Socher. (2015, April). CS 224D: Deep Learning for NLP Lecture Notes: Part I. https://cs224d.stanford.edu/lecture_notes/LectureNotes1.pdf

(4) Godfried, I. (s. f.). ICML 2018: Advances in transfer, multitask, and semi-supervised learning. June 1st, 2018, de https://towardsdatascience.com/icml-2018-advances-in-transfer-multitask-and-semi-supervised-learning-2a15ef7208ec

(5) John G. Breslin, Parsa Ghaffari. (2019, February). Neural Transfer Learning for Natural Language Processing. Sebastian Ruder. https://ruder.io/thesis/neural_transfer_learning_for_nlp.pdf

(6) Kevin Clark1 Minh-Thang Luong2 Christopher D. Manning1 Quoc V. Le. (2018a). Semi-Supervised Sequence Modeling with Cross-View Training. arXiv, 1-34.https://arxiv.org/pdf/1809.08370.pdf

(7) Levine, S. (2019). Transfer and Multi-Task Learning [Diapositivas].http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_15_multi_task_learning.pdf

(8) Li, D., & Li, X. (2013, May). Machine Learning Paradigms for Speech Recognition: An Overview. May 2nd, 2020, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.337.8867&rep=rep1&type=pdf#

(9) Modern Deep Learning Techniques Applied to Natural Language Processing by Authors. (s. f.). June 13rd, 2020, de https://nlpoverview.com/#1

(10) Ruder, S. (s. f.). Transfer Learning. June 1st, 2020, de https://ruder.io/thesis/neural_transfer_learning_for_nlp.pdf#page=60##Related Tasks used in NLP

(11) Singh, S. (2018). Natural Language Processing for Information Extraction. arXiv, 12-40. https://arxiv.org/pdf/1807.02383.pdf

(12) Small, S. (s. f.). Review of Information Extraction Technologies and Applications. May 19, 2020, https://www.researchgate.net/publication/259146203_Review_of_Information_Extraction_Technologies_and_Applications

(13) Soni, D. (s. f.). Supervised vs. Unsupervised Learning. March 19, 2020, https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d

(14) Text-to-speech (TTS) Overview. (s. f.). June 3rd, 2020, http://www.voicerss.org/tts/default.aspx

(15) Vargas, E. (s. f.). A Comprehensive Introduction to Word Vector Representations. June 3rd, 2020, https://medium.com/ai-society/jkljlj-7d6e699895c4

(16) Victor Sanh, Thomas Wolf, Sebastian Ruder. (2018b, noviembre 26). A Hierarchical Multi-task Approach for Learning Embeddings from Semantic Tasks. February 19, 2020, https://arxiv.org/abs/1811.06031

(17) Python | shutil.copy() method. (s. f.). March 1, 2020, https://www.geeksforgeeks.org/python-shutil-copy-method/

(18) Shah, T. (s. f.). About Train, Validation and Test Sets in Machine Learning. June 9, 2020, https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7

(19) Train, Validation and Test Sets. (December 3rd, 2017). Recuperado de https://tarangshah.com/blog/2017-12-03/train-validation-and-test-sets/

(20) Brownlee, J. (2019, August 8th). A Gentle Introduction to k-fold Cross-Validation. https://machinelearningmastery.com/k-fold-cross-validation/

(21) Cross-validation: evaluating estimator performance. (s. f.). https://scikit-learn.org/stable/modules/cross_validation.html

(22) Training spaCy's Statistical Models · spaCy Usage Documentation. (s. f.). June 9, 2020, https://spacy.io/usage/training/

(23) S. Thrun and L. Pratt, Eds., Learning to learn. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

(24) R. Caruana, "Multitask learning," Machine Learning, vol. 28(1), pp. 41– 75, 1997.

(25) M. M. H. Mahmud and S. R. Ray, "Transfer learning using kolmogorov complexity: Basic theory and empirical evaluations," in Proceedings of the 20th Annual Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press, 2008, pp. 985–992.

(26) E. Eaton, M. desJardins, and T. Lane, "Modeling transfer relationships between learning tasks for improved inductive transfer," in Machine Learning and Knowledge Discovery in Databases, European Confer- ence, ECML/PKDD 2008, ser. Lecture Notes in Computer Science. Antwerp, Belgium: Springer, September 2008, pp. 317–332.

(27) M. T. Rosenstein, Z. Marx, and L. P. Kaelbling, "To transfer or not to transfer," in a NIPS-05 Workshop on Inductive Transfer: 10 Years Later, December 2005.

(28) Learning representations by back-propagating errors. David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams (1988).

(29) What makes a gene name? Named entity recognition in the biomedical literature. Ulf Leser and Jo¨rg Hakenberg. 21st July 2005 Henry Stewart Publications 1467-5463. Briefings In Bioinformatics. Vol 6. No 4. 357–369. December 2005