ParTNER: Paragraph Tuning for Named Entity Recognition on Clinical Cases in Spanish using mBERT + Rules

Antonio Tamayo[1], Diego Burgos[2] and Alexander Gelbukh[1]

[1] *Instituto Politécnico Nacional (IPN), Centro de Investigación en Computación (CIC), Av. Juan de Dios Batiz, s/n, 07320, Mexico City, Mexico*
[2] *Wake Forest University, 1834 Wake Forest Road, Winston-Salem, NC 27109, Winston Salem, USA*

Abstract

Named entity recognition (NER) and normalization are crucial tasks for information extraction in the medical field. They have been tackled through different approaches from rule-based systems and classic machine learning methods with feature engineering to the most sophisticated deep learning models; most of them for English. In this work, we present a transfer learning approach starting from multilingual BERT to tackle the problem of Spanish NER (species) and normalization in clinical cases by using sentence tokenization for training and a paragraph tuning strategy at the inference phase. We propose that text lengths at training and inference stages do not have to match and that such difference can leverage the model's performance according to the task. Our validation showed that using a context of three sentences during inference improves the F1 score in ≈1% compared to longer and shorter paragraphs and in ≈17% compared to the whole document. We also applied simple but effective post-processing rules on the model's output, which improved the Micro F1 score in ≈28%. Our system achieved an F1 of 0.8499 in the testing dataset of the LivingNER shared task 2022.

Keywords

Named entity recognition, normalization, transfer learning, multilingual BERT, paragraph tuning

1. Introduction

Named entities in a scientific field are part of the specific knowledge of the domain. All together, disease mentions, drug interactions, species mentions, etc. make up the layer of specialized knowledge in a document set that could and should be reusable in more than one specific application [1]. The way this knowledge is processed, acquired, learned, and reused has changed dramatically in the last decade. While ontologies, knowledge-based systems, and grammars were key resources during the initial stages of specialized text processing, current modern deep learning algorithms trained on big data seem to be able to represent and encapsulate the complexity of this knowledge system into reusable, interoperable, language models.

In this paper, we report an automatic system to identify and delimit species mentions in clinical cases and to annotate each mention with a taxonomy identifier from the NCBI (National Center for Biotechnology Information) [2] in a Spanish version (https://doi.org/10.5281/zenodo.6390506). These experiments are a contribution to tasks 1 and 2 of LivingNER 2022: Named entity recognition, normalization and classification of species, pathogens, and food [3]. Task 1 is particularly challenging because the evaluation metric measures the system's ability to determine the exact starting and ending location of the species mention in the document, while Task 2's main challenge was the uncertainty of not knowing whether all the species mentions detected by our system actually were recorded in the huge NCBI taxonomy.

For Task 1, we followed a transfer learning approach, that is, we used and fine-tuned multilingual BERT (mBERT), a language model that was originally trained on different, more heterogeneous data,

IberLEF2022, September 2022, A Coruña, Spain

i.e., not only medical documents, and that was also trained for a different task. Besides fine-tuning the built-in hyperparameters of the model (learning rate, epochs, etc.), we propose an additional hyperparameter to determine the best text length during the inference phase, which we call ParTNER (Paragraph Tuning for Named Entity Recognition). A multi-sentence text length (ParTNER = 3), which we describe below, yields better results than longer and shorter paragraphs and than complete documents. We think this hyperparameter takes into account mBERT's difficulty to handle long texts [4] and it also means that training and inference text lengths do not necessarily have to match in this model and for this specific task.

Since we addressed Task 1 as a sequence labeling problem, the training data provided by the organizers required substantial preprocessing in order to take it to the format expected by the model. Likewise, we carried out simple but effective post-processing on the model's output in order to concatenate subwords, clean up the extracted entities, and take the predictions to the format required by LivingNER. This postprocessing proved to be very useful to improve the system's precision (Micro F1 = 0.8499). For Task 2, we take Task's 1 output and map the detected mentions onto the NCBI taxonomy, which gave promising results.

The paper is structured as follows. Section two presents relevant works related to NER and transfer learning in specific domains. Section three describes the methodology including the dataset, preprocessing steps, the language model and fine-tuning used, as well as the ParTNER hyperparameter, postprocessing, and the mapping of the NCBI codes onto the recognized entities. Section four reports and analyzes the results, and Section five draws some conclusions on the present work.

2. Related Works

Natural language processing tasks in the medical field have been typically addressed with one of the following three approaches or a combination of them: a) rule-based, b) machine learning feature engineering, or c) deep learning techniques. As for the rule-based approach, [5] reports promising results in English for NER using rules and regular expressions. On the other hand, the first works utilizing machine learning techniques to tackle NER in the medical field reached their best results using feature engineering, support vector machines (SVMs) and tree-based kernels [6].

Deep learning has recently become a popular approach for NER in medical documents, particularly for tasks such as disease mention detection and drug interaction identification. For example, [7] used BERT [8] and ELMO [9] in two corpora in English, namely, PubMed titles and abstracts and clinical notes. They used a BIO scheme [10] and reached their highest F1 score (86.6) on disease mention detection with a BERT-base model on PubMed only using the configurations reported by [8]. [11] carried out transfer learning for biomedical NER in English also. They pre-trained a bidirectional language model (BiLM) on unlabeled data and used it to initialize weights instead of initializing them randomly. This strategy outperformed other experiments without pre-training or with unidirectional pre-training. Their F1 scores reached 87.34 on the NCBI-Disease corpus and 89.28 on the BC5CDR corpus. It is worth noting that besides disease mentions, BC5CDR includes chemical entities, which seem to respond better to bidirectional models. (cf. [7]). In [12, 13], the authors re-trained BERT on a huge biomedical corpus in English and then fine-tuned it for different tasks including NER; [12] accomplished an impressive improvement (0.51%) in biomedical NER. Likewise, [14] trained BERT but reports better results fine-tuning the model presented in [12] than their own. With regards to data sets with interconnected documents, LinkBERT [15] sets state-of-the-art performance for biomedical NER tasks by pre-training BERT on a corpus of PubMed abstracts and citation links to scientific papers. PubMedBERT [16] is also pretrained from scratch on PubMed abstracts and full-text articles. Both systems often outperform other systems in the medical domain, the former by taking advantage of training on independent but related documents, and the latter by pre-training directly on biomedical documentation only.

On the other hand, named entity normalization has also been addressed using deep learning, such as CNN [17] or transformer-based methods [18]. In [18], the authors compared the results achieved by fine-tuning BERT, BioBERT [12], and ClinicalBERT [19] for normalization in three datasets, and they showed an improvement of 1.17% in comparison with the state-of-the-art.

Regarding experiments with Spanish, [20] published a corpus with nested entities and addressed the NER problem using a biLSTM-CRF architecture and word embeddings trained over both clinical embeddings and Spanish Wikipedia. In [19], the authors proposed a novel join deep learning model to tackle NER and normalization in a corpus of cancer in Spanish achieving an F1 score of 0.87 on NER and an F1 of 0.825 on normalization; an equivalent result was reported by [21, 22] using ensemble pre-trained BERT models and post-processing.

With regards to deep learning and text length supported by language models, it has been reported that BERT does not get along well with long texts [4]. Some systems such as Longformer [23] and Big Bird [24] contribute an adaptation of transformers so they scale linearly instead of quadratically with document length. This allows these systems to train with long texts although their computation cost is high, which is why we were unable to use them in the present work with the available technical resources. Alternatively, approaches like sliding windows of varying lengths for training and inference have been used in tasks such as question answering and summarization [25]. In the legal domain, [26] split long documents into paragraphs to fine tune BERT for legal case retrieval.

3. Methodology

3.1. Dataset

The dataset for these tasks is one of the first and most important contributions to the NLP community in Spanish. It consists of 2,000 clinical cases in Spanish in plain, unstructured text on one hand, and annotation files on the other comprising the character offsets of the entity mentions together with their corresponding NCBI Taxonomy code annotations. Species and Human mentions in the original clinical cases were manually annotated by experts following thorough guidelines and inter-annotator agreement. These annotations were used to generate the structured files, which have the following fields (see also Table 1):

- *filename*: document name
- *mark*: identifier mention id
- *label*: mention type (SPECIES or HUMAN)
- *off0*: starting position of the mention in the document
- *off1*: ending position of the mention in the document
- *span*: text span
- *NCBI* code

Table 1
Dataset sample for the NER and normalization tasks

filename	mark	label	off0	off1	span	NCBI code
file_1	T1	SPECIES	2132	2142	SARS-CoV-2	2697049
file_1	T2	SPECIES	1781	1792	antivíricos	10239
file_1	T3	HUMAN	75	93	médico de cabecera	9606
file_1	T4	HUMAN	3	11	paciente	9606

These clinical cases are reports from about 20 medical subdisciplines ranging from cardiology to radiology and dermatology, which makes it a robust representation of the specialized knowledge of the field. Out of the 2,000 clinical cases, the organization randomly chose 1,000 for the training set, 500 for the development or validation set, and 500 for the test set. In order to minimize the chance of bias, the test set, which has no annotations, was shuffled with a large background corpus of additional clinical cases, but the tasks were evaluated only on the 500 documents originally chosen for the test set.

3.2. System Description

In this work, we present a transformer-based system enhanced with simple post-processing to tackle the recognition and normalization of species in clinical cases in Spanish. Our proposal involves four stages, namely, pre-processing, fine-tuning, paragraph tuning, and post-processing. Additionally, we map the recognized mentions onto NCBI codes as our contribution to LivingNER Task 2. Below we describe each of them.

3.2.1. Pre-processing

Instead of the popular BIO tagging scheme (Begin: B; Inside: I; Outside: O), we used IO and adapted it to our own scheme so we could address the NER problem as a sequence labeling one using the labels "S" (SPECIES) and "H" (HUMAN) instead of "I" and "O" (Outside). Simply using IO (in or out of entity mentions) tags suffices with BERT models, leading to similar or better performance compared to BIO tags and more complex variants [16]. Besides, an exploratory observation of LivingNER data showed few cases of contiguous entities, so we saw no need to put more burden on the model by adding a *Beginning* tag. This required pre-processing the data provided by the organizers of LivingNER to transform the original data (see section 3.1) to the SHO format. During the pre-processing, it was necessary to tokenize and align each lexical unit to its corresponding label as shown in the example below.

"La[O] madre*[H]* refirió[O] sensación[O] de[O] congestión[O] y[O] tos[O] durante[O] los[O] últimos[O] días[O] y[O] un[O] hermano*[H]* con[O] síntomas[O] víricos*[S]* . La[O] madre*[H]* tiene[O] antecedentes[O] de[O] lesiones[O] herpéticas*[S]* orales[O] recurrentes[O] ."

With regards to dedicated tools for this task, despite recent development of resources for NER such as a biomedical SciSpacy [27] for English, tools and resources for Spanish are still limited. Therefore we used Spacy [28] instead to tokenize the documents. Likewise, although tokenization can be considered a straightforward task, aligning some tokens, particularly punctuation marks, to their labels still pose multiple problems. The pre-processing was simple but tedious and time-consuming.

3.2.2. Fine-tuning mBERT

Multilingual BERT is a version of BERT [8], a language model trained to predict masked words and next sentence on a large corpus in 104 languages, including Spanish. Because we tackled the NER problem as a sequence labeling one, the model's head of prediction was changed for the fine-tuning process.

We tried three different approaches of text splitting to fine-tune the model, namely, whole documents, paragraphs (maximum 5 sentences), and sentence tokenizing. For the last two approaches, sentences were split by the rule "period plus space". As it will be shown later, sentence tokenizing was the best option for this stage. For the fine-tuning process, we used random partitions of the training dataset into training (75%) and validation (25%) sets. It was done iteratively five times with random seeds, and the results that we present in Table 3 are an average of the results obtained in these iterations. Additionally, we carried out a hyperparameter tuning searching for the best model's configuration using a grid search for the epochs (3, 5, 7) and the learning rate (5e-03, 5e-05, 5e-07). 7 epochs and a learning rate of 5e-05 yielded the best results. We kept the default values for the rest of hyperparameters. For this process, we used the transformer library and the mBERT cased model available at Hugging Face (https://huggingface.co/bert-base-multilingual-cased). Google Colab Pro with a GPU Tesla P100 with 27.3 gigabytes of available RAM was used to run all the experiments. The clean version of the data we used for our training process together with the source code to replicate this work are available at a GitHub repository (https://github.com/ajtamayoh/NLP-CIC-WFU-Contribution-to-LivingNER-shared-task-2022).

3.2.3. ParTNER

Because mBERT does not have the ability to work with large documents, we propose ParTNER (paragraph tuning for NER), an additional hyperparameter at the inference phase consisting of the length of the text that the model needs to maximize correct named entity predictions. For this purpose, we used the same sentence tokenization from the fine-tuning phase and tested with a paragraph approach with a different number of sentences, namely, 1, 2, 3, 5, 7, and the whole document. The effect of ParTNER's various values is shown later in the Results section.

3.2.4. Post-processing

Simple post-processing was carried out through a custom Python script to clean up and format the output. Firstly, because mBERT works with a subword tokenization system, we decode the output that contained subwords. Secondly, we concatenate all the named entities detected by the model one after the other. This means that if the model detects a named entity whose final character position (or final character position plus one) concurs with the first position of the next named entity detected, our system considers that these two entities are part of one single entity. This was necessary because the model extract parts of some entities separately. Thirdly, we also applied simple but effective post-processing based on some orthographic and grammatical rules which are detailed in Table 2. Lastly, since we work with the SHO scheme, the last post-processing step consisted of decoding the predictions to put them in the data format required by LivingNER, which was described in section 3.1. Likewise, under the assumption that LivingNER participants were required to extract all the mentions of an entity occurring in the same clinical case, we used the entities extracted by the model to identify and extract any repetitions of said entities in the same document.

Table 2
Post-processing rules

If the named entity detected …	… then apply this rule
Starts with punctuation mark	Delete the match and fix the off0
Contains a mark of new line	Replace the match with a space
Contains a space before and/or after a hyphen or a parenthesis	Delete the space(s) and fix the off1
Ends with non-content words or punctuation marks	Delete the match and fix the off1
Concurs with non-content words or punctuation marks	Leave out of the entities detected

3.2.5. Task 2: Mapping NCBI codes

We followed a quite simple approach to address the LivingNER's normalization task. First, taking advantage of the NCBI taxonomy, which the organizers translated into Spanish, we built a dictionary

linking this taxonomy with all the entities and its NCBI codes in the competition's training and validation datasets. Then, having the output for the NER task, we looked up in this dictionary the corresponding code for each named entity previously extracted by our system. If the code matched an entry in the dictionary, it was used to annotate the entity. This explains why we report the same results on our validation dataset for both tasks. The promising results obtained with this simple approach suggest that task 2 is highly dependent on the results of task 1.

4. Results and analysis

In this section, we present the results achieved by our system on the training, validation, and test datasets. For all these results, the model's hyperparameter configuration is as mentioned above. The metrics used to present the results are micro precision (MiP), micro recall (MiR), and micro F1 (MiF1). Table 3 shows the results for the fine-tuning phase with the training dataset using different text splitting approaches, without any post-processing, and under the IO scheme previously explained.

Table 3
Average and standard deviation for Task 1 - NER on the training dataset

Model	Text splitting approach	Micro Precision	Micro Recall	Micro F1
mBERT	Whole document	0.8999 +/- 0.0021	0.6973 +/- 0.0020	0.7857 +/- 0.0004
	Paragraph	0.8701 +/- 0.0065	0.7187 +/- 0.0048	0.7872 +/- 0.0055
	Sentence	0.8734 +/- 0.0112	0.7209 +/- 0.0130	0.7897 +/- 0.0056

Table 3 shows why we ended up using the sentence for training, as it slightly outperforms whole documents and paragraphs (maximum 5 sentences). However, based on the standard deviation, the performance of the three approaches can be considered similar. The evaluation on this training set aimed at determining the best hyperparameter configuration for Task 1 with mBERT. For this preliminary evaluation we used our own SHO annotations as gold standard rather than the starting and ending locations of entity mentions in LivingNER's data. This is why we do not carry out a thorough error analysis on this stage yet. The results in Table 3 do not include post-processing.

Once the pre-trained mBERT had been fine-tuned for species mention detection in Spanish, we ran our predictions on the validation dataset for both NER and normalization tasks. Table 4 shows the results achieved in the inference phase by setting ParTNER to 3, that is a context of three sentences, and by passing the post-processing rules. For Task 2, we carried out the NCBI code mapping described above.

Table 4
Results for Task 1 - NER and Task 2 - Normalization on the validation dataset

Model	ParTNER	Micro Precision	NER & Norm Micro Recall	Micro F1
mBERT + post-processing	1	0.8277	0.8938	0.8595
	2	0.8309	0.8898	0.8593
	3	0.8426	0.8877	**0.8646**
	5	0.8429	0.8756	0.8589
	7	0.8390	0.8538	0.8463
	Whole document	0.8689	0.5763	0.6930

Table 4 shows that using a context of three sentences during inference improves the F1 score in ≈1% compared to longer and shorter paragraphs and in ≈17% compared to the whole document. Likewise, it is worth noting that we reached the best results using a different text length during the training and inference phases. For training, we used individual sentences and, for inference, the best results were obtained with a ParTNER of 3 sentences. The result obtained using whole documents was expected

because mBERT cannot process long texts due to its quadratically increasing memory and time consumption [4].

In order to illustrate the effect of the post-processing rules on the final results, Table 5 shows the scores obtained for both NER and normalization tasks on our validation dataset using the best ParTNER hyperparameter but without passing the rules detailed in Table 2.

Table 5
Results for Task 1 - NER and Task 2 - Normalization on the validation dataset without rules

Model	Paragraph length	Micro Precision	NER & Norm Micro Recall	Micro F1
mBERT	3	0.4439	0.8594	0.5854

The impact of the postprocessing rules on the final results is remarkable. The rules fix some of the model's inaccuracies that affect precision. As we describe in the error analysis below, the model does a good job identifying the entity mentions, but not so good delimiting the exact location of many of them. This is not surprising at all since the task of accurately delimiting a term in context, or a species mention in this case, is hard, not only for machines but also for human experts. [29], for example, proved that human experts are good at identifying approximate windows of specialized mentions in context, but they do not perform as well to determine the starting and ending position of the term. Additionally, some extracted units, which we call false entities (e.g., non-content words only, punctuation marks, etc.) impact the Micro Precision significantly. Notice that we report the same results on the validation dataset for NER and normalization tasks because of the reasons that we explained in section 3.2.5.

Table 6 shows our system's results reported by LivingNER's organizers for both tasks on the test set. A more pronounced decrease is observed in the system's performance for the normalization task (0.8646 → 0.7951) than for the NER task (0.8646 → 0.8499). This was expected due to the simple mapping method employed to tackle the normalization task and because we did not train the system to annotate composite mentions (i.e., several NCBI taxonomy codes were required to map a single annotated mention) or to use terminology codes that are more general than the annotated mention, which were features of the competition's dataset. It is also likely that many entities in the test set were not in the dictionary that we created to map the NCBI codes. The decrease in the NER task scores was also expected, although the change is not dramatic compared to our validation results and the system's performance for this task is promising.

Table 6
Results for Task 1 - NER and Task 2 - Normalization on the test dataset

Model	ParTNER	NER MiP	NER MiR	NER MiF1	Norm MiP	Norm MiR	Norm MiF1
mBERT + post-processing	3	0.8303	0.8704	**0.8499**	0.7768	0.8143	**0.7951**

4.1. Error analysis

We provide here a brief qualitative analysis of the model's errors so the reader can grasp 1) what the model learns and predicts and 2) the usefulness of the postprocessing rules. Table 7 lists examples of the errors that we have identified. We have classified these errors into six types, which are described below in the table.

Table 7
Error examples

Gold standard				Predicted		
id	**off0**	**off1**	**span**	**off0**	**off1**	**span**
1	2924	2953	Salmonella tiphy y parathiphy	2924	2953	Salmonella tiphy y parathiphy
2	2924	2940	Salmonella tiphy	N/A	N/A	Not predicted
3	N/A	N/A	Not an entity	3455	3461	raquis
4	N/A	N/A	Not an entity	3884	3890	vector
5	3090	3095	virus	3090	3116	virus VIH, hepatitis B y C
6	3096	3099	VIH	N/A	N/A	Not predicted
7	3101	3116	hepatitis B y C	N/A	N/A	Not predicted
8	3026	3043	influenza A FC, B	3026	3046	influenza A FC, B FC
9	955	976	bacilos grampositivos	955	989	bacilos grampositivos pleomórficos
10	N/A	N/A	Not an entity	908	916	positivo
11	N/A	N/A	Not an entity	967	975	positivo
12	N/A	N/A	Not an entity	1081	1089	positivo
13	N/A	N/A	Not an entity	2946	2960	microorganismos
14	1928	1965	Corynebacterium pseudo--diphtheriticum	1928	1950	Corynebacterium pseudo
15	1138	1161	C. pseudodiphtheriticum	1141	1161	pseudodiphtheriticum

- **Nested entities (**examples 1 and 2**):** This is a common phenomenon in sequence labeling problems, which causes a decrease in the recall metric since our system does not extract the nested entities separately, but the larger entity only.
- **False entities** (examples 3 and 4): This a type of false positives. They are meaningful entities that the model extracts, but they are from a different domain or category. This error affects the precision of the model.
- **Expansion** (examples 5 to 9): The model extracts the named entity but it adds some text to it, which may be part of another entity (example 5) or a false entity (examples 8 and 9). Notice that examples 6 and 7 show two false negatives which are a product of the expansion error in example 5. These types of errors affect both the recall and precision of the model.
- **Propagation** (examples 10 to 13): Propagation errors caused by our postprocessing strategy to extract any repetitions in the same clinical case of an entity extracted by the model (see 3.2.4 above). They affect the precision of the model.
- **Pre-processing** (example 14): This error may be due to the tokenization carried out as part of the pre-processing to take the data to the SHO scheme. We may have not tokenized non-alphanumeric characters properly, which may have misled the model. This type of error generates false positives and false negatives also affecting both the precision and recall.
- **Omission** (example 15): The model truncates an entity. This error also affects the precision and the recall of the system.

5. Conclusions

In this work, we report our contribution to LivingNER 2022. We present a competitive system to tackle Task 1, that is, the recognition of species mentions in clinical cases in Spanish. The straightforward methodology followed for Task 2, NCBI code annotation, can be considered a strong baseline although it depends on the output of Task 1, which incorporates more sophistication than

expected for a baseline. The code and data to replicate our experiments have been made available at GitHub.

Our system achieved a micro F1 of 0.8499 for NER and a micro F1 of 0.7951 for the normalization task. Although the system uses a well-known fine-tuning technique for mBERT, the chosen hyperparameter configuration, the post-processing rules, and the ParTNER hyperparameter proposed leveraged the model's performance. The post-processing rules boosted the model's precision and the ParTNER hyperparameter applied at the inference phase increased recall of the model. We carried out a qualitative error analysis that shed some light into the model's behavior and which should be taken into account for future work in order to enhance a system with this configuration. Some of these errors, such as false entities, expansion, or propagation, can be addressed with additional rules to improve the system's precision, whereas nested entities, pre-processing, and omission errors affect recall and require higher level approaches.

6. Acknowledgements

This work was done with partial support from the Mexican Government through the grant A1-S-47854 of CONACYT, Mexico, grants 20220852 and 20220859 of the Secretaría de Investigación y Posgrado of the Instituto Politécnico Nacional, Mexico. The authors thank the CONACYT for the computing resources brought to them through the Plataforma de Aprendizaje Profundo para Tecnologías del Lenguaje of the Laboratorio de Supercómputo of the INAOE, Mexico and acknowledge the support of Microsoft through the Microsoft Latin America PhD Award.

7. References

[1] Varadarajan S, Kasravi K, Feldman R. Text-mining: Application development challenges. Applications and innovations in intelligent systems X. 2003:247-60.

[2] choch, Conrad L et al. "NCBI Taxonomy: a comprehensive update on curation, resources and tools." Database : the journal of biological databases and curation vol. 2020 (2020): baaa062. doi:10.1093/database/baaa062

[3] Antonio Miranda-Escalada, Eulàlia Farré-Maduell, Salvador Lima-López, Darryl Estrada, Luis Gascó and Martin Krallinger. Mention detection, normalization & classification of species, pathogens, humans and food in clinical documents: Overview of LivingNER shared task and resources. Procesamiento del Lenguaje Natural. 2022.

[4] Ding M, Zhou C, Yang H, Tang J. Cogltx: Applying bert to long texts. Advances in Neural Information Processing Systems. 2020;33:12792-804.

[5] Eftimov T, Koroušić Seljak B, Korošec P. A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. PloS one. 2017 Jun 23;12(6):e0179488.

[6] Patra R, Saha SK. A kernel-based approach for biomedical named entity recognition. The Scientific World Journal. 2013 Jan 1;2013.

[7] Peng Y, Yan S, Lu Z. Transfer learning in biomedical natural language processing: an evaluation of BERT and ELMo on ten benchmarking datasets. arXiv preprint arXiv:1906.05474. 2019 Jun 13.

[8] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018 Oct 11.

[9] Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L. Deep contextualized word representations. arXiv 2018. arXiv preprint arXiv:1802.05365. 1802;12.

[10] Ramshaw LA, Marcus MP. Text chunking using transformation-based learning. In Natural language processing using very large corpora 1999 (pp. 157-176). Springer, Dordrecht.

[11] Sachan DS, Xie P, Sachan M, Xing EP. Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition. In Machine learning for healthcare conference 2018 Nov 29 (pp. 383-402). PMLR.

[12] Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics. 2020 Feb 15;36(4):1234-40.
[13] Akhtyamova L. Named entity recognition in Spanish biomedical literature: Short review and bert model. In 2020 26th Conference of Open Innovations Association (FRUCT) 2020 Apr 20 (pp. 1-7). IEEE.
[14] Alsentzer E, Murphy JR, Boag W, Weng WH, Jin D, Naumann T, McDermott M. Publicly available clinical BERT embeddings. arXiv preprint arXiv:1904.03323. 2019 Apr 6.
[15] Yasunaga M, Leskovec J, Liang P. LinkBERT: Pretraining Language Models with Document Links. arXiv preprint arXiv:2203.15827. 2022 Mar 29.
[16] Gu Y, Tinn R, Cheng H, Lucas M, Usuyama N, Liu X, Naumann T, Gao J, Poon H. Domain-specific language model pretraining for biomedical natural language processing. ACM Deng P, Chen H, Huang M, Ruan X, Xu L. An ensemble CNN method for biomedical entity normalization. InProceedings of the 5th workshop on BioNLP open shared tasks 2019 Nov (pp. 143-149).
[17] Ji Z, Wei Q, Xu H. Bert-based ranking for biomedical entity normalization. AMIA Summits on Translational Science Proceedings. 2020;2020:269.
[18] Si Y, Wang J, Xu H, Roberts K. Enhancing clinical concept extraction with contextual embeddings. Journal of the American Medical Informatics Association. 2019 Nov 1;26(11):1297-304.
[19] Báez P, Villena F, Rojas M, Durán M, Dunstan J. The Chilean Waiting List Corpus: a new resource for clinical named entity recognition in Spanish. InProceedings of the 3rd clinical natural language processing workshop 2020 Nov (pp. 291-300).
[20] Xiong Y, Huang Y, Chen Q, Wang X, Nic Y, Tang B. A joint model for medical named entity recognition and normalization. Proceedings http://ceur-ws. org ISSN. 2020;1613:0073.
[21] García-Pablos A, Perez N, Cuadros M. Vicomtech at cantemist 2020. InProceedings of the Iberian Languages Evaluation Forum (IberLEF 2020), CEUR Workshop Proceedings 2020.
[22] Wang Z, Ng P, Ma X, Nallapati R, Xiang B. Multi-passage bert: A globally normalized bert model for open-domain question answering. arXiv preprint arXiv:1908.08167. 2019 Aug 22.
[23] Beltagy I, Peters ME, Cohan A. Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150. 2020 Apr 10.
[24] Zaheer M, Guruganesh G, Dubey KA, Ainslie J, Alberti C, Ontanon S, Pham P, Ravula A, Wang Q, Yang L, Ahmed A. Big bird: Transformers for longer sequences. Advances in Neural Information Processing Systems. 2020;33:17283-97.
[25] Grail Q, Perez J, Gaussier E. Globalizing BERT-based transformer architectures for long document summarization. InProceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume 2021 Apr (pp. 1792-1810).
[26] Shao Y, Mao J, Liu Y, Ma W, Satoh K, Zhang M, Ma S. BERT-PLI: Modeling Paragraph-Level Interactions for Legal Case Retrieval. InIJCAI 2020 Jul (pp. 3501-3507).
[27] Neumann M, King D, Beltagy I, Ammar W. ScispaCy: fast and robust models for biomedical natural language processing. arXiv preprint arXiv:1902.07669. 2019 Feb 20.
[28] Honnibal M, Montani I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. 2017 Jul;7(1):411-20.
[29] Estopà, R, Martí, J, Burgos, D, Luna, J, Monserrat, S, Montané, A, Muñoz, P, Quispe, W, Rivadeneira, M, Rojas, E, Sabater, M, Salazar, H, Samara, A, Santis, R, Seghezzi, N, Fernández, S, Souto, M. La identificación de unidades terminológicas en contexto: de la teoría a la práctica. 2006:1000-30.