When Specialization Helps: Using Pooled Contextualized Embeddings to Detect Chemical and Biomedical Entities in Spanish

Manuel Stoeckel
Goethe University Frankfurt
Text Technology Lab
manuel.stoeckel@stud.uni-frankfurt.de

Wahed Hemati
Goethe University Frankfurt
Text Technology Lab
hemati@em.uni-frankfurt.de

Alexander Mehler
Goethe University Frankfurt
Text Technology Lab
mehler@em.uni-frankfurt.de

Abstract

The recognition of pharmacological substances, compounds and proteins is an essential preliminary work for the recognition of relations between chemicals and other biomedically relevant units. In this paper, we describe an approach to Task 1 of the PharmaCoNER Challenge, which involves the recognition of mentions of chemicals and drugs in Spanish medical texts. We train a state-of-the-art BiLSTM-CRF sequence tagger with stacked Pooled Contextualized Embeddings, word and sub-word embeddings using the open-source framework FLAIR. We present a new corpus composed of articles and papers from Spanish health science journals, termed the *Spanish Health Corpus*, and use it to train domain-specific embeddings which we incorporate in our model training. We achieve a result of 89.76% F1-score using pre-trained embeddings and are able to improve these results to 90.52% F1-score using specialized embeddings.

1 Introduction

Efficient access to information on chemicals and pharmaceutical units has become increasingly important for researchers in various chemical disciplines. However, manual annotation of these units to create knowledge bases is a laborious process given the ever-increasing number of papers and patents in bio/chemical and pharmaceutical research. Thus, *Natural Language Processing* (NLP) can be employed to detect such entities and their relations from the relevant literature. Previous work has been successful in detecting and classifying chemical substances or in extracting complex relations between chemical substances (Krallinger et al., 2015; Hemati and Mehler, 2019).

While most NLP research is conducted on English datasets, there are a considerable number of non-English biomedically relevant texts written in other languages, e.g. clinical texts. In order to advance the further development of biomedical and pharmaceutical entity recognition facing this linguistic diversity, the PharmaCoNER task challenges participants with *Named Entity Recognition* (NER) for pharmacological substances, compounds and proteins on a Spanish corpus (Gonzalez-Agirre et al., 2019b). The PharmaCoNER task belongs to the *BioNLP Open Shared Tasks 2019* (BioNLP-OST 2019) Workshop and distinguishes two tracks: the first track focuses on NER offset and entity classification, while the second task deals with concept indexing.

In this paper we present an architecture for NER of chemical and pharmacological units in Spanish texts that produces an F-score of up to 90%. Source code and instructions for reproducing these results are available on GitHub[1] and we are offering an interactive web service for testing our models.[2] The article is organized as follows: First, we describe the resources used to train our model and explain our methodical approach. This includes a detailed description of the PharmaCoNER dataset and the kind of preprocessing we performed on the input texts. Afterwards, we give a thorough description of our architecture. Finally, we discuss our results and give our conclusions.

2 Materials and Methods

2.1 Datasets

In this section, we describe the datasets used in our experiments and the architecture of our NER tagger.

[1] www.git.io/JenqE

[2] espharmaner.texttechnologylab.org

Proceedings of the 5th Workshop on BioNLP Open Shared Tasks, pages 11–15
Hong Kong, China, November 4, 2019. ©2019 Association for Computational Linguistics

PharmaCoNER The corpus accompanying the PharmaCoNER task, that is, the *Spanish Clinical Case Corpus* (SPACCC), contains 1 000 manually classified clinical cases and comprises 396 988 token (Gonzalez-Agirre et al., 2019a). The corpus was derived from open access Spanish medical publications and (according to the creators) shows properties of both biomedical and medical literature as well as clinical records.

The SPACCC corpus is given in brat standoff format[3] as two separate files per document, one containing the plain text, the other containing the annotations with character level offsets on the raw text. We converted the corpus into a CoNLL2003 compatible format, applying common whitespace tokenization and splitting tokens on non-alphanumeric characters, as this increased the performance of our model.

Spanish Health Corpus In this section, we describe the Spanish Health Corpus, a collection of 7353 diverse Spanish health and science journal articles and papers. The corpus was obtained from SciELO[4] by means of an automated crawler.[5] The content of the articles in this corpus was downloaded as embedded text from the respective websites and stripped of any structural elements, like HTML tags. Then, the raw text was split into sentences using DEEP-EOS, a neural network sentence boundary detection tool created by Stefan Schweter which is publicly available on GitHub.[6]

We trained a Spanish DEEP-EOS LSTM model on 100 000 Spanish Wikipedia sentences extracted from the Leipzig Corpora Collection (Goldhahn et al., 2012). Our DEEP-EOS model achieves an accuracy of 99.65% on separate 100 000 test sentences. The resulting sentences were then tokenized based on the procedure mentioned in the previous section. This resulted in a set of 957 648 sentences containing 32 346 137 words in total. We used this corpus to train special word embeddings for our system that we believe have a positive impact on the performance of our models.

2.2 System Architecture

Our system was built with FLAIR (Akbik et al., 2019a), an easy to use open-source NLP framework that is able to produce state-of-the-art results for sequence tagging tasks (eg. Akbik et al., 2018, 2019b). We follow the approach of Akbik et al., using FLAIR to stack (i.e. concatenate) character and word embeddings to improve recognition rates. We further expand this model by adding sub-word embeddings to the stacked embeddings. These stacked embeddings serves as input for a BiLSTM-CRF sequence tagger (Huang et al., 2015).

For our best performing model, we used two different token-level embeddings, a WANG2VEC-based embedding (Ling et al., 2015) and a FASTTEXT-based embedding (Bojanowski et al., 2017), a single byte-pair sub-word embedding (Heinzerling and Strube, 2018) and one context sensitive character-level language model (Akbik et al., 2019b). Figure 1 gives a visual depiction of our best performing model. The following paragraphs describe the used embeddings in more detail.

Pooled Contextualized Embeddings *Contextualized String Embedding*s (Akbik et al., 2018, CSEs) use pre-trained character-level language models from which hidden states at the start and end character positions of each word are extracted to create embeddings for any string in sentence contexts. This model is further developed by Akbik et al. (2019b) who introduce an expansion to CSEs in terms of *Pooled Contextualized Embeddings* (PCEs).

PCEs tackle the problem of embedding rare words by applying a pooling operation on different contextual embeddings of the word. The authors follow the idea that words which occur in underspecified contexts should be familiar to the reader from previous mentions. So when a word is processed during the training of a character-level language model, all previous contextualized instances of the word are pooled and concatenated with the current instance to create a "global" word representation (Akbik et al., 2019b). The authors experiment with three pooling operations (*min*, *max* and *mean*). In this way, they are able to achieve state-of-the-art results in four major NER tasks (Akbik et al., 2019b).

In our architecture, we employ pre-trained Spanish Pooled Contextualized Embeddings.[7]

wang2vec Embeddings This model, proposed by Ling et al. (2015), is an extension of the token-

[3] brat.nlplab.org/standoff.html

[4] www.scielo.org

[5] See our GitHub repository for the list of documents.

[6] www.github.com/stefan-it/deep-eos

[7] These models were trained by Yihwa Kim (www.github.com/iamyihwa).

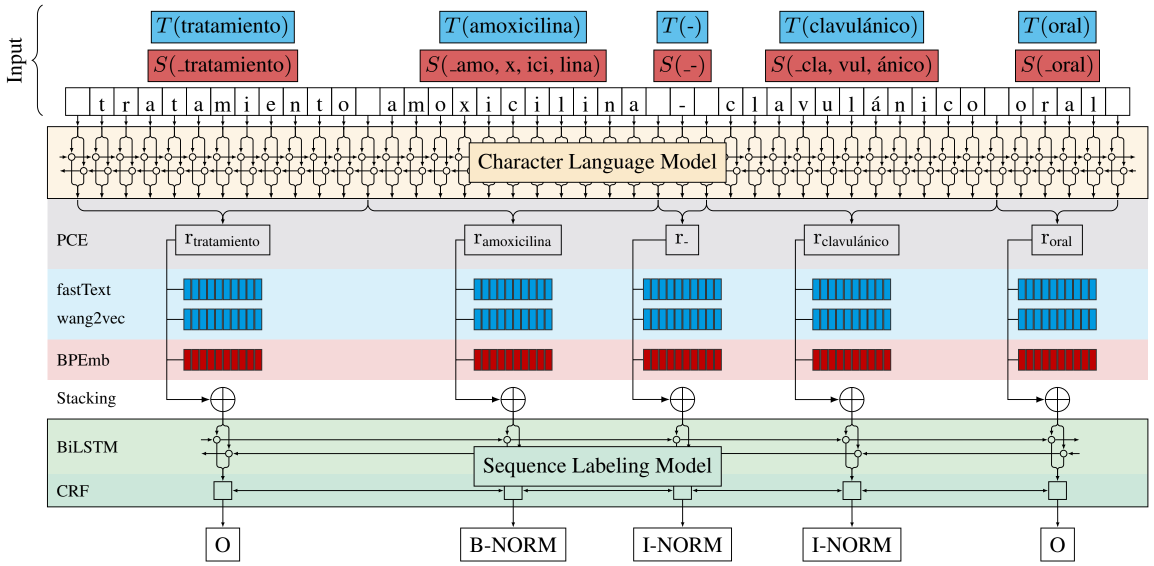

Figure 1: The architecture of the best performing model in our experiments. The PCEs are generated from C character features, while FASTTEXT and WANG2VEC embeddings are trained on T tokens, and BPEMB uses S syllable input. The embeddings are stacked and serve as input for a BiLSTM-CRF Sequence Labeling Model.

level WORD2VEC model of Mikolov et al. (2013). During training, WANG2VEC makes a prediction for each neighboring position of the target word instead of making a single prediction for all neighbours. Thus, the resulting embeddings are better at capturing syntactic, positional information (Ling et al., 2015).

We trained 300 dimensional WANG2VEC-based embeddings based on 100 iterations using default parameters on the Spanish Health Corpus.

fastText Embeddings Unlike WORD2VEC or WANG2VEC, FASTTEXT (Bojanowski et al., 2017) models words as sets of character n-grams, where all n-grams from sizes 3-6 are used during training. FASTTEXT can thus represent rare words that were not present in the vocabulary of the training files if their skip-grams were observed during training. Before the words are split into n-grams, special boundary symbols are added. The embeddings are thus also able to learn information about word prefixes and suffixes (Bojanowski et al., 2017). We used pre-trained 300 dimensional Spanish FASTTEXT embeddings from Grave et al. (2018) in our initial submission to the PharmaCoNER task.[8]

We replaced them with our own 300 dimensional embeddings trained on the Spanish Health Corpus with standard parameter settings during our experimental phase.

Byte-Pair Embeddings. Similar to FASTTEXT, Byte-Pair embeddings (Heinzerling and Strube, 2018, BPEMB) are trained on a pre-processed corpus that contains sub-word entities. But in contrast to FASTTEXT, words in the training corpus are represented as combinations of *syllables* instead of skip-grams. These syllables or *subword units* are learned from the corpus prior to the segmentation using Byte-Pair-Encoding (Sennrich et al., 2016) for a predefined number.

In our experiments, we used pre-trained 300 dimensional Spanish Byte-Pair embeddings made available by Heinzerling and Strube (2018) with a syllable vocabulary size of 100 000.[9]

2.3 Experiments

We conducted extensive experiments to optimize our models. The ease of use of FLAIR enables us to swap embeddings and optimizers on the fly and perform a state-of-the-art hyper-parameter search. Following the “best known configurations” for NER tasks in English, German and Dutch according Akbik et al.’s GitHub repository,[10] we trained the BiLSTM-CRF sequence tagger with a hidden size of 256, a single LSTM layer (unless stated otherwise) and no dropout. We used common *Stochastic Gradient Descent* (SDG) with a learning rate of 0.1, mini-batch size of 32, an annealing rate of 0.5 with a patience of 3 and default parame-

[8] www.fasttext.cc/docs/en/crawl-vectors.html

[9] www.github.com/bheinzerling/bpemb

[10] www.github.com/zalandoresearch/flair

ters otherwise. The training takes about 80 epochs with these settings.

We performed a parameter search with FLAIR's wrapper of the hyper parameter selection tool HYPEROPT (Bergstra et al., 2013). We chose our initial search parameters similar to the search conducted by Akbik et al. (2019b), which includes a learning rate $\in \{0.01, 0.05, 0.1\}$ and mini-batch size $\in \{8, 16, 32\}$. Using this parameter set we were unable to improve our models performance over the performance using the suggested ones. In addition, we ran a sparse parameter search with a different array of possible choices: hidden size $\in \{256, 512\}$, dropout $\in [0, 0.5]$, number of RNN layers $\in \{1, 2\}$ and learning rate $\in \{0.05, 0.1, 0.15\}$. While all of the trained models performed very well, we were unable to outperform our previous best model.

All experiments were performed either on a NVIDIA GTX 1660 with 6 GiB VRAM available or on a NVIDIA GTX 1080 Ti with 11 GiB VRAM available.

3 Evaluation

Results Table 1 compares the scores of our systems. All scores were computed using the official evaluation script provided by the organizers of the PharmaCoNER task on the gold standard test data, which was released after the end of the challenge phase. After establishing a baseline using mean-pooled PCEs only, we added pre-trained Byte-Pair embeddings (BPEMB-PRE) and pre-trained FASTTEXT (FT-PRE) embeddings. While Byte-Pair embeddings alone were able to increase the performance of the model by $+3.61\%$ F1-score, further adding pre-trained FASTTEXT embeddings only increased the systems performance about $+0.11\%$ for a total of $+3.72\%$ against our baseline. This confirms the observations of Akbik et al. (2019b) according to which stacking token-level embeddings on PCEs can improve the performance of the model significantly. Adding a second LSTM layer to the BiLSTM sequence tagger decreased the models F1-score by 0.54% as can be seen in the second entry in row 3 of table 1.

After the challenge phase, we replaced the pre-trained FASTTEXT embeddings with our self-trained, specialized embeddings (FT^S) and added the specialized WANG2VEC (W2V^S) embeddings. This increased the performance of the system to 90.34% F1-score. The choice of mean-pooled PCEs in favor of min-pooled PCEs resulted in a further increase in performance to 90.52% F1-score, representing a total increase of $+4.48\%$ over our baseline and $+0.76\%$ over our best result during the challenge phase, while choosing max-pooled PCEs results in the highest precision score of all our models (90.97%).

Model	F1-Score	Precision	Recall
PCE-PRE (BSE)	86.04	88.59	83.64
BSE + BPEMB-PRE† (SBM)	89.65	90.45	88.86
SBM + FT-PRE			
1 LSTM layer†‡	89.76	90.69	88.85
2 LSTM layers†	89.22	89.10	89.34
SBM + FT^S + w2v^S			
min-pooled*	90.31	90.02	89.71
max-pooled*	90.34	**90.97**	89.71
mean-pooled*	**90.52**	90.79	**90.30**

Table 1: All scores in %. BSE denotes our baseline, while SBM denotes our first submission model. The notation "X + Y" is to be read as "X stacked with Y". Legend: † indicates challenge submissions,
‡ indicates the best challenge submission,
^{S} indicates self-trained specialized embeddings,
* indicates models built after the challenge deadline.

4 Conclusion and Future Work

Our experiments show that with current frameworks like FLAIR it is possible to achieve very good test results with little time spent on system development or implementation. Good results can be achieved with pre-trained models and embeddings that are available in many languages thanks to the NLP community's ongoing efforts.

Our experiments confirm our expectations regarding the usability of special embeddings. The embeddings that are trained on the Spanish Health Corpus contribute to significantly increasing the performance of our system, even with such a small training corpus. Our results show that the use of domain-specific embeddings can significantly improve the performance of sequence tagging models even in the case of small corpora.

We will be continuing our experiments in due time, using larger corpora for our training. In the mean time all our results, datasets and code necessary to reproduce our experiments have been made publicly available on GitHub and can be tested with an interactive web service.

Acknowledgements

We would like to thank the anonymous reviewers for their fair opinions and the organisers for their patience and help with problems during the submission of the workshop papers.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019a. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019b. Pooled contextualized embeddings for named entity recognition. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 724–728.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

J. Bergstra, D. Yamins, and D. D. Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages I–115–I–123. JMLR.org.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12*.

Aitor Gonzalez-Agirre, Ander Intxaurrondo, and Jose Antonio Lopez-Martin. 2019a. Description of the corpus for the PharmaCoNER challenge. `http://temu.bsc.es/pharmaconer/index.php/description-of-the-corpus/`, [Accessed: 01.08.2019].

Aitor Gonzalez-Agirre, Montserrat Marimon, Ander Intxaurrondo, Obdulia Rabal, Marta Villegas, and Martin Krallinger. 2019b. Pharmaconer: Pharmacological substances, compounds and proteins named entity recognition track. In *Proceedings of the BioNLP Open Shared Tasks (BioNLP-OST)*, pages 1–X, Hong Kong, China. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Wahed Hemati and Alexander Mehler. 2019. LSTMVoter: chemical named entity recognition using a conglomerate of sequence labeling tools. *Journal of Cheminformatics*, 11(1):7.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. Cite arxiv:1508.01991.

Martin Krallinger, Florian Leitner, Obdulia Rabal, Miguel Vazquez, Julen Oyarzabal, and Alfonso Valencia. 2015. Chemdner: The drugs and chemical names extraction challenge. *Journal of cheminformatics*, 7(1):S1.

Wang Ling, Chris Dyer, Alan Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Neural and Information Processing System (NIPS)*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725. Association for Computational Linguistics (ACL).