Task-Aware Contrastive Pre-training for Spanish Named Entity Recognition in LivingNER Challenge

Sumam Francis[1,*], Marie-Francine Moens[1]

[1]KU Leuven, Belgium
[1]KU Leuven, Belgium

Abstract

This paper is presented as part of the LivingNER-Species NER track (Species mention entity recognition). The goal of the task is to identify the character offsets of all species mentions (human or non-human) in a collection of plain text clinical case reports. In this paper, we explore the idea of incorporating contrastive pre-training followed by fine-tuning for this task. Contrastive pre-training is performed such that it increases the divergence between non-related entities and decrease the divergence between related entities. The model first trains the sentence in the text, and calculate the similarity between token categories based on their Gaussian-distributed embeddings in the sentences. Further, this contrastively pre-trained BERT model is combined with the classification head to perform the named entity recognition task. We demonstrate that fine-tuning a model pre-trained using the contrastive loss performs better than directly fine-tuning on the clean, annotated data. The models have been evaluated on the IberLEF 2022 challenge using the official release of the LivingNER corpus [1]. Experimental results shows that our model outperforms baseline methods on the task.

Keywords

Contrastive Learning, Named Entity Recognition, Information Extraction, Gaussian embeddings

1. Introduction

Information extraction involves extracting structured data from unstructured text. This is applied in many fields like recommender systems, retrieval systems, sentiment analysis and biomedical information extraction. Named Entity Recognition (NER) is the most important component of the information extraction system. NER aims to identify named entities from unstructured texts belonging to different label categories. With the advent of large pre-trained models like BERT [2], RoBERTa [3], BART [4], Xlnet [5], representations learned by such models achieve strong performance across many tasks in NLP. The architectures of these models are mostly based on Transformers [6] which uses a self-attention mechanism to capture long-range dependencies between tokens. Lately most NER systems make use of a fine-tuning approach on these transformer-based pre-trained models yielding state of the art results on many NER bench-marking tasks [2].

Many recent works make use of training in two phases: pre-training and further fine-tuning. Pre-training is a process in which a model is first trained on an auxiliary task before performing

IberLEF 2022, September 2022, A Coruna, Spain

[*] Corresponding author.

sumam.francis@kuleuven.be (S. Francis); sien.moens@kuleuven.be (M. Moens)

a final fine-tuning phase on target task. The motivation behind pre-training is that learning an auxiliary task first will help capture relevant information and the representations learned during the pre-training phase can be reused in the supervised fine-tuning of the downstream task by adapting the architecture and parameters to the downstream task.

Recent work [7] [8] [9] [10] showed that downstream performance can be improved by further adapting a general pre-trained model by continued pre-training on more relevant set of downstream tasks. The representations learned in the task adaptive pre-training (TAPT) [8] involves pre-training using an unsupervised objective on not just the similar domain but the actual end-task and dataset itself. It has shown to improve the performance of the model for the target task and is less computationally demanding.

Recently, supervised contrastive learning has become popular. Contrastive learning is a representation learning method, which has been widely used for visual [11] and text representations [12]. It builds positive pairs between samples with the same class label and puts their representations together. The main idea is to learn a representation by contrasting positive and negative sample pairs. Specifically, it puts positive pairs together and pushes negative pairs away.

This paper reports on a submission to the LivingNER shared task. For the Living NER shared task, we incorporate the idea of task adaptive pre-training together with supervised contrastive pre-training loss. For the NER task, label information is used to contrastively learn to push apart different token categories and pull together similar token categories. In comparison to conventional contrastive learning that optimizes the similarity objective between point embedding representations [13] [14], we optimize the distributional divergence of Gaussian embeddings similar to [15]. While point embedding representation only optimize for the sample distances, Gaussian embeddings also include additional constraint of preserving the class distribution through its variance estimates and at the same time improve generalization [16][17][15]. We demonstrate the effectiveness of contrastive learning in discriminating between different token categories through our experiments on the LivingNER NER dataset.

Given the importance of mixing data augmentations [18] during contrastive pre-training we utilize the multilingual datasets provided as part of the shared task to obtain sufficient different views of the data. Since incorporating text augmentations that alter the meaning (semantics) of a sentence can adversely affect the performance of the model, we restrict ourselves to translations provided as part of the shared task challenge.

- We explore a supervised contrastive pre-training approach for NER which uses Gaussian embeddings to optimize the distributional divergence. We demonstrate the effectiveness of contrastive learning in contrasting between different token categories.
- We demonstrate that representations learned from multilingual data augmentations generated from translations improve the contrastive clustering and learning.
- The model is evaluated on the living NER shared task validation set and test set for NER sub_task. Our contrastively pre-trained model performs better than the baseline BERT model.

2. Method

Our model uses contrastive learning based supervised pre-training to maximize distributional divergence between representations of different tokens. This method improves the model's ability to categorize and differentiate between different token categories. Furthermore, modeling Gaussian embedding instead of conventional point representation effectively lets the model learn the generalized entity class distribution. This contrastively pre-trained model is further fine-tuned for the named entity recognition task by adding a classification head. Finally, it lets us fine-tune our model even with a small number of samples without over-fitting which is imperative for domain adaptation.

2.1. Model architecture

A BERT [2] based pre-trained language model encoder is used to generate contextualized representation of sentence tokens. Given a sequence of n tokens $X = [x_1, x_2, ..., x_n]$, the encoder layer f_θ maps it into a sequence of hidden vectors. The final hidden layer representations of BERT encoder are used as the intermediate representations $h = \{h_0, h_1, ..., h_n\}$.

The intermediate representations are further passed through a linear projection layer to create the embedding representations. Here instead of using point embedding representations for calculating the contrastive loss, we use Gaussian embedding representations as in [8]. The token embedding representations are assumed to follow Gaussian distributions. Two linear projection layers are used to model the mean ($f_{\boldsymbol{\mu}}$) and covariance ($f_{\boldsymbol{\Sigma}}$) parameters of the Gaussian distribution.

$$\boldsymbol{\mu}_i = f_{\boldsymbol{\mu}}(h_i), \boldsymbol{\Sigma}_i = ELU(f_{\boldsymbol{\Sigma}}(hi)) + (1 + e) \tag{1}$$

$\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ represent the mean and covariance of the Gaussian embeddings respectively. $f_{\boldsymbol{\mu}}$ and $f_{\boldsymbol{\Sigma}}$ are implemented as ReLU followed by single layer MLP. ELU refers to exponential linear unit and $e \approx e^{-14}$ which is added for numerical stability.

2.2. Supervised Contrastive pre-training

To perform contrastive pre-training on the dataset we make use of the contrastive loss. The KL divergence between all valid token pairs are taken to calculate the contrastive loss. Valid token pairs include all tokens in the sentence which are not special tokens or padding tokens.

Two tokens x_a and x_b are positive samples if they have same labels $y_a = y_b$. $\mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)$ represents the Gaussian embeddings of token x_a and $\mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b)$ represents the Gaussian embeddings of token x_b. Given Gaussian embeddings $\mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)$ and $\mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b)$, the KL divergence is calculated as below:

$$
\begin{aligned}
D_{\mathrm{KL}}\left[\mathcal{N}_b \| \mathcal{N}_a\right] &= D_{\mathrm{KL}}\left[\mathcal{N}\left(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b\right) \| \mathcal{N}\left(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a\right)\right] \\
&= \frac{1}{2}\Big(\operatorname{Tr}\left(\boldsymbol{\Sigma}_a^{-1} \boldsymbol{\Sigma}_b\right) \\
&\quad + \left(\boldsymbol{\mu}_a - \boldsymbol{\mu}_b\right)^T \boldsymbol{\Sigma}_a^{-1}\left(\boldsymbol{\mu}_a - \boldsymbol{\mu}_b\right) \\
&\quad - l + \log \frac{\left|\boldsymbol{\Sigma}_a\right|}{\left|\boldsymbol{\Sigma}_b\right|}\Big)
\end{aligned} \tag{2}
$$

where Tr refers to the trace operator. Due to KL-divergence not being symmetric, both directions of the KL-divergence are calculated and their average is used as the contrastive loss.

$$
d(a, b) = \frac{1}{2}\left(D_{\mathrm{KL}}\left[\mathcal{N}_b \| \mathcal{N}_a\right] + D_{\mathrm{KL}}\left[\mathcal{N}_a \| \mathcal{N}_b\right]\right) \tag{3}
$$

At each training step, during contrastive pre-training, a batch of sequences X_{sub} are randomly sampled without replacement from the training set X. For each token $(x_a, y_a) \in X_{sub}$, we obtain its Gaussian Embedding $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ by passing the corresponding token sequence through the model [15]. The in-batch positive samples X_a for sample a are selected and subsequently calculate the contrastive loss of X_a with respect to that of all other valid tokens in the batch.

$$
X_a = (x_b, y_b) \in X_{sub} \mid y_a = y_b, a \neq b \tag{4}
$$

We can then calculate the distributional divergence of all the token pairs in the batch.

$$
\ell(a) = -\log \frac{\sum_{(x_b, y_b) \in X_a} \exp(-d(a, b)) / \left|X_a\right|}{\sum_{(x_b, y_b) \in X, a \neq b} \exp(-d(a, b))} \tag{5}
$$

For all token pairs $i \in (x_i, y_i) \in X_{sub}$ calculate the $l(i)$.

$$
\mathcal{L}_{contr} = \frac{1}{|X_{sub}|} \sum_{i \in X_{sub}} \ell(i) \tag{6}
$$

Further reduce $\mathcal{L}_{contr}$ by updating $f_{\boldsymbol{\mu}}, f_{\boldsymbol{\Sigma}}$ and encoder representations through back-propagation.

2.3. Fine-tuning and inference

After performing task specific contrastive pre-training, the next step is to fine-tune the model on data specifically for NER task. In [13] they find that the representations before the projection layers are more informative than the representations obtained from the Gaussian embedding projection heads. Hence the output representation from the encoder part of the contrastively pre-trained model is extracted.

$$
o = f_\theta(x_i) \tag{7}
$$

Then, a multi-layer perceptron (MLP) classifier followed by the softmax function is added to obtain the label probability distribution of x_i:

$$
p(x_i, \theta) = softmax(W.o_i + b) \tag{8}
$$

where W and b are the weight and bias terms. The model is then trained in a supervised manner by minimizing the Cross entropy loss of predicted probability and the ground truth label.

3. Experimental Setup

3.1. Implementation details

We implemented the models based on HuggingFace's Transformer [19] with PyTorch [20] in one single NVIDIA Titan RTX GPU. We used pretrained models XLM-R and Spanish BERT model as the base encoder models from Huggingface library [19].

Then, we explored further task adaptive pre-training of these 2 models on the translations provided as part of the challenge and further fine-tune on the Spanish training set. For the task of adaptive contrastive pre-training we trained the models for 20 epochs with learning rate of $5e - 5$ and maximum sequence length of 128. We kept the Gaussian embedding dimension fixed to 128. The training loss is the KL divergence loss.

For the fine-tuning step, we used the encoder output representations of the contrastively pre-trained model. Epochs, batch size, maximum sequence length, learning rate, and optimizer were set to $20, 32, 128, 5e - 5$, and AdamW [21] respectively. Each experiment was repeated 3 times and the average entity-level micro average Precision, Recall and F1-score are reported. The target models are evaluated on the development set (25% of training set provided was taken as development set) every 750 steps. The checkpoints are saved based on the F1 scores obtained on the development set. The training loss for fine-tuning is the cross entropy loss.

3.2. Dataset

The LivingNER Gold Standard training set is composed of 1000 clinical case reports extracted from miscellaneous 20 medical specialties including covid, oncology, infectious diseases, tropical medicine, urology, pediatrics, and others. For LivingNER-Species NER sub-task, annotations are provided in a tab-separated file (TSV) file. The clinical case reports are annotated manually for species [SPECIES] and [HUMAN] entities.

The validation set is composed of 500 clinical cases from many different specialties: covid, oncology, infectious diseases, tropical medicine, urology, allergology, etc. The test+background set is a collection of 13467 clinical case reports. The goal of the LivingNER task is to develop automatic systems for Spanish medical texts by making predictions for the test+background set. Among the 13467 clinical case reports, 485 are used for evaluation (this is the test set). The rest (background set) are added to prevent manual annotations and to create a silver standard.

Named entity recognition tagging methods mainly include IOB, BIO, BIOES etc. BIO requires that all named entities start with a 'B' tag, 'I' refers to inside of the named entity, and 'O' refers to not a named entity. The labeling scheme used in this dataset is BIO tagging.

Table 1
Micro-average Precision (P),Recall (R) and F1 scores (F1) on the validation set of the Living NER shared task.

Model	Precision	Recall	F1
SPANISH-BERT	0.9654	0.9562	0.9608
Contr-SPANISH-BERT	0.9691	0.9596	0.9643
XLM-R	0.9730	0.9538	0.9633
Contr-XLM-R	0.9786	0.9688	0.9680

Table 2
Micro-average Precision (P), Recall (R) and F1 scores (F1) on the test set of the Living NER shared task.

Model	LivingNER NER			NER only SPECIES			NER only HUMAN		
	P	R	F1	P	R	F1	P	R	F1
Contr-SPANISH-BERT	0.9448	0.9299	0.9373	0.9238	0.9038	0.9137	0.9740	0.9651	0.9695
Contr-XLM-R	0.9443	0.9307	0.9375	0.9232	0.9047	0.9139	0.9738	0.9657	0.9697

3.3. Evaluation metrics

The main evaluation metric for NER task is the F1 score. There are two types of F1 score, namely macro-averaged F1 score, and micro-averaged F1 score. The macro-average F1 score calculates the F1 score for each entity category separately, and then calculates the overall average. The micro-average F1-score computes the F1-score for all the test instances and avarages this score. For this task we evaluate the micro averaged F1 score, precision and recall scores for two named entities HUMAN and SPECIES. The evaluations are performed using the official evaluation script provided by the LivingNER shared task [22].

4. Results and Discussion

Table 1 compares the various methods investigated in this paper on the validation dataset of the LivingNER shared sub_task. In general, the proposed contrastively pre-trained model performs better than the baseline method. The results indicate the effectiveness of the explored method in this paper.

It can be seen from the Table 1 that contrastively pre-trained xlm_roberta model achieves the best performance on the validation set of LivingNER shared task. The next best performance is obtained using the contrastively pre-trained Spanish BERT model.

Contrastively pre-training the model maximizes the distributional divergence of tokens during pre-training and provides a good starting point for fine-tuning for a few epochs to achieve better NER recognition compared to baseline methods. Moreover KL-divergence between Gaussian embeddings have shown to be effective in explicitly considering the asymmetric distance which better represents similarity between structurally similar words.

We submitted the best performing models to LivingNER task for evaluation on the test+Background set. Table 2 shows the results of our task adaptive contrastively pre-trained methods on the test set of the LivingNER shared sub_task. From the Table 2, we see that contrastively pre-trained XLM-R performs slightly better than contrastively pre-trained Spanish BERT model which can be accounted by the larger size of the XLM-R model.

5. Conclusion

In this paper, we described our submissions for the NER sub_task of the LivingNER competition. We explored contrastive learning based supervised task adaptive pre-training framework that maximizes the inter token divergence by modelling Gaussian embeddings. By utilizing supervised contrastive task adaptive pre-training, we were able to improve upon the already high-performing transformer-based models. We achieved a overall F1-score of 0.9375 on the LivingNER task.

6. Acknowledgments

The authors would like to thank the text mining group at the Barcelona supercomputing center for the smooth organization of LivingNER. This research was financed by the AIDA (AI tools for DAta curation of clinical registries) project (KU Leuven C3/20/117).

References

[1] A. Miranda-Escalada, E. Farré-Maduell, S. Lima-López, D. Estrada, L. Gascó, M. Krallinger, Mention detection, normalization and classification of species, pathogens, humans and food in clinical documents: Overview of livingner shared task and resources, Procesamiento del Lenguaje Natural (2022).

[2] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, Association for Computational Linguistics, 2019. doi:10.18653/v1/n19-1423.

[3] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019).

[4] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in: D. Jurafsky, J. Chai, N. Schluter, J. R. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.acl-main.703.

[5] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, in: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (Eds.), Advances in Neural Information

Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS, 2019.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, 2017.

[7] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinform. 36 (2020) 1234–1240. URL: https://doi.org/10.1093/bioinformatics/btz682.

[8] S. Gururangan, A. Marasovic, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, N. A. Smith, Don't stop pretraining: Adapt language models to domains and tasks, in: D. Jurafsky, J. Chai, N. Schluter, J. R. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, Association for Computational Linguistics, 2020, pp. 8342–8360. URL: https://doi.org/10.18653/v1/2020.acl-main.740.

[9] I. Beltagy, K. Lo, A. Cohan, Scibert: A pretrained language model for scientific text, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, Association for Computational Linguistics, 2019, pp. 3613–3618. URL: https://doi.org/10.18653/v1/D19-1371. doi:10.18653/v1/D19-1371.

[10] L. M. Dery, P. Michel, A. Talwalkar, G. Neubig, Should we be pre-training? an argument for end-task aware training as an alternative, 2021. URL: https://arxiv.org/abs/2109.07437. doi:10.48550/ARXIV.2109.07437.

[11] J. Carse, F. A. Carey, S. J. McKenna, Unsupervised representation learning from pathology images with multi-directional contrastive predictive coding, CoRR abs/2105.05345 (2021). URL: https://arxiv.org/abs/2105.05345. arXiv:2105.05345.

[12] T. Gao, X. Yao, D. Chen, Simcse: Simple contrastive learning of sentence embeddings, in: M. Moens, X. Huang, L. Specia, S. W. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, Association for Computational Linguistics, 2021, pp. 6894–6910. URL: https://doi.org/10.18653/v1/2021.emnlp-main.552. doi:10.18653/v1/2021.emnlp-main.552.

[13] T. Chen, S. Kornblith, M. Norouzi, G. E. Hinton, A simple framework for contrastive learning of visual representations, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of *Proceedings of Machine Learning Research*, PMLR, 2020, pp. 1597–1607. URL: http://proceedings.mlr.press/v119/chen20j.html.

[14] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, D. Krishnan, Supervised contrastive learning, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/

d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html.

[15] S. S. S. Das, A. Katiyar, R. J. Passonneau, R. Zhang, Container: Few-shot named entity recognition via contrastive learning, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, Association for Computational Linguistics, 2022, pp. 6338–6353. URL: https://aclanthology.org/2022.acl-long.439.

[16] C. Qian, F. Feng, L. Wen, T. Chua, Conceptualized and contextualized gaussian embedding, in: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, AAAI Press, 2021, pp. 13683–13691. URL: https://ojs.aaai.org/index.php/AAAI/article/view/17613.

[17] B. Athiwaratkun, A. G. Wilson, Hierarchical density order embeddings, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018. URL: https://openreview.net/forum?id=HJCXZQbAZ.

[18] R. Zhou, Q. Hu, J. Wan, J. Zhang, Q. Liu, T. Hu, J. Li, WCL-BBCD: A contrastive learning and knowledge graph approach to named entity recognition, CoRR abs/2203.06925 (2022). URL: https://doi.org/10.48550/arXiv.2203.06925. doi:10.48550/arXiv.2203.06925.

[19] huggingface transformers library, https://github.com/huggingface/transformers, 2022. Accessed: 01-07-2022.

[20] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, in: NIPS-W, 2017.

[21] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net, 2019. URL: https://openreview.net/forum?id=Bkg6RiCqY7.

[22] A. Miranda-Escalada, Livingner evaluation library, https://github.com/tonifuc3m/livingner-evaluation-library, 2022. Accessed: 01-07-2022.