RACAI at SemEval-2022 Task 11: Complex named entity recognition using a lateral inhibition mechanism

Vasile Păiș
Research Institute for Artificial Intelligence ”Mihai Drăgănescu”, Romanian Academy
Bucharest, Romania
vasile@racai.ro

Abstract

This paper presents RACAI's system used for the shared task of "Multilingual Complex Named Entity Recognition (MultiCoNER)", organized as part of "The 16th International Workshop on Semantic Evaluation (SemEval 2022)". The system employs a novel layer inspired by the biological mechanism of lateral inhibition. This allowed the system to achieve good results without any additional resources apart from the provided training data. In addition to the system's architecture, results are provided as well as observations regarding the provided dataset.

1 Introduction

Named entity recognition (NER) is a well known task in natural language processing. It aims to detect spans of text associated with known entities. Initially, much work focused on detecting persons, organizations, and locations (Grishman and Sundheim, 1996; Tjong Kim Sang and De Meulder, 2003). However, this limited approach is not suitable for every domain, thus leading to research in domain-specific NER. For example, in the biomedical domain, a number of works have addressed entities such as genes, proteins, diseases (Hu and Verberne, 2020), cell types (Settles, 2004), chemicals (Gonzalez-Agirre et al., 2019; Ion et al., 2019). Similarly, in the legal domain additional classes are employed such as money value (Glaser et al., 2018), legal reference (Landthaler et al., 2016; Păiș et al., 2021), judge, and lawyer (Leitner et al., 2019).

In the context of "The 16th International Workshop on Semantic Evaluation (SemEval 2022)"[1], the task number 11 "Multilingual Complex Named Entity Recognition (MultiCoNER)"[2] (Malmasi et al., 2022b) required participants to build a NER system able to recognize complex entities in 11 languages: Bangla, Chinese, Dutch, English, Farsi, German, Hindi, Korean, Russian, Spanish, and Turkish. In addition, a multilingual track and a code-mixed track were available. The task focused on 6 entity types: person, location, group, corporation, product and creative work.

As noted by Ashwini and Choi (2014), non-traditional entities can pose a challenge for NER systems. This happens because datasets are harder to build and certain entities (such as creative works) are updated more frequently than traditional ones (persons, locations). Furthermore, traditional entities tend to occur as noun phrases, while the newly proposed entities (for the purposes of the task) may be linguistically complex (complex noun phrases, gerunds, infinitives or full clauses). An interesting result was provided by Aguilar et al. (2017), where the top system from WNUT 2017 achieved only 8% recall when dealing with creative works.

This paper describes a system for complex NER in a multilingual context, developed at the Research Institute for Artificial Intelligence of the Romanian Academy (RACAI), that participated in the MultiCoNER task. The system employs a new artificial neural network layer trying to mimic the biological process of *lateral inhibition* (Cohen, 2011). In various regions of the brain, excited neurons can reduce the activity of other neighbouring neurons. In the visual cortex this process may account for an increased perception in low-lighting conditions. Thus, intuitively the newly proposed system may better focus on subtle details present in the data and the language model.

The paper is structured as follows: Section 2 presents related work, Section 3 describes the dataset and pre-processing operations used, Section 4 describes the method used with the system architecture in Section 4.1 and performed experiments in Section 4.2. The results are given in Section 5 and finally, conclusions and future work

[1] https://semeval.github.io/SemEval2022/

[2] https://multiconer.github.io/

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1562 - 1569
July 14-15, 2022 ©2022 Association for Computational Linguistics

are available in Section 6.

2 Related work

In a survey regarding NER using deep learning models, Yadav and Bethard (2018) emphasize the importance of pre-trained word embedding representations. Dernoncourt et al. (2017), in the NeuroNER package[3], combine pre-trained word embeddings with character level embeddings passing through a neural network with a final CRF layer achieving high scores on different datasets. Other authors, using similar architectures have shown that combining multiple static word representations can further increase the overall system performance (Păiș and Mitrofan, 2021).

With the introduction of contextual word representation models, such as BERT (Devlin et al., 2019), ELMo (Peters et al., 2018), ROBERTA (Liu et al., 2019), XLNet (Yang et al., 2019), NER systems have been adapted to make use of these new models. For the majority of the contextual models, depending on the size of the artificial neural network being used, we distinguish between a base version and a large version (with a larger number of parameters). Devlin et al. (2019) used the BERT-large model for NER on the CoNLL-2003 English dataset, achieving an F1 score of 92.8%. Nguyen et al. (2020) propose a custom BERT-like model for English tweets, called BERTweet, for improving the NER performance on two datasets: WNUT-16 and WNUT-17.

Contextualized embeddings were also applied with success in domain-specific settings. Considering the ProfNER shared task (Miranda-Escalada et al., 2021), dedicated to identifying mentions of occupations in health-related social media, the best performing system made use of the BETO model (Cañete et al., 2020) trained on a large Spanish corpus.

Wang et al. (2021) propose an algorithm for automatically finding a concatenation of embeddings that improves a system's performance in different tasks, including NER. The authors considered multiple contextual and non-contextual embeddings and the proposed algorithm can identify any of their combinations. Their work builds on previous experiments that showed increased performance when manually concatenating contextual and non-contextual embeddings (Straková et al., 2019; Wang et al., 2020).

Training a contextualized embedding model requires many processing resources. This means that not all flavours are available for all languages. Multilingual models have been proposed, making use of training data in multiple languages. Such models include mBERT and XLM-RoBERTa (XLM-R) (Conneau et al., 2020). These models are known to perform especially well on low resourced languages. Considering NER, Conneau et al. (2020) show that XLM-R large performs better than mBERT, providing an average 2% increase in F1 score for Dutch, Spanish and German. Interestingly, in English the performance of XLM-R is more similar to mBERT (though still offering an improved performance of less than 1%) and less than (Akbik et al., 2018). This seems to support the idea that monolingual models, trained on a specific language or domain, can perform better than multilingual ones.

Meng et al. (2021) recognizes the importance of gazetteer resources, even in the case of state-of-the-art systems making use of contextualized embeddings. The authors propose using an encoder for obtaining Contextual Gazetteer Representations (CGRs) as a way to incorporate any number of gazetteers into a single, span-aware, dense representation. Then, the authors go one step further and propose a gated Mixture-of-Experts (MoE) method to fuse CGRs with contextual word representations from any word-level model.

Fetahu et al. (2021) employ multilingual gazetteers fused with transformer models in a MoE approach to improve the recognition of entities in code-mixed web queries. In this case the entities were written in a different language than the rest of the query, thus posing particular challenges to existing NER systems.

3 Dataset and pre-processing

The dataset (Malmasi et al., 2022a) was provided in a column-based format, with 4 columns in each file. The text was tokenized, with tokens available in the first column. Columns 2 and 3 did not contain useful information (only an underscore symbol was present). Column 4 contains the named entity annotation in BIO format. The first token (or the single token) of an entity contains the "B-" prefix. Other entity tokens (in the case of multi-token entities) start with an "I-" prefix, while non-entity tokens are denoted with "O".

For each language there was a train/dev/test split

[3] http://neuroner.com/

provided by the organizers. In the train set there were 15,300 sentences, in the dev set there were 800 sentences, while the test set was much larger. The actual number of sentences in the test set varies between languages, having as many as 217,818 sentences for English. For the multilingual track, there were 168,300 train sentences, 8,800 dev sentences and 471,911 test sentences. The smallest number of sentences was given in the code-mixed track with only 1,500 training sentences, 500 dev sentences and 100,000 test sentences.

Regardless of the language or additional multilingual and code-mixed tracks the development set is very small while the test set is much larger than the training set. In addition, the number of training sentences in the code-mixed track seems insufficient to build a model by themselves.

Predictions on the test set were required to provide annotations, in the same BIO format, for each token. Furthermore, considering the multilingual nature of the task, with very different languages (such as German and Chinese), no pre-processing was applied, other than converting the provided format to a format suitable for the developed system. For the code-mixed track, a second dataset was generated, based on the original provided dataset augmented with new sentences. These were generated by randomly replacing existing entities in the code-mixed dataset with similar entities extracted from the multilingual dataset. For each sentence containing at least one entity, a new sentence was generated, thus doubling the size of the training dataset for the code-mixed track.

4 Method

4.1 System Architecture

The proposed system employs the XLM-RoBERTa (Conneau et al., 2020) large model. Given an input sequence from the dataset it is first transformed into model-specific tokens. The sequence is also enhanced with the special tokens for sequence start (CLS), sequence end (SEP) and, if needed, padding (PAD). Then it passes through the model to obtain associated contextual embeddings. In existing systems, the word representations usually pass through a linear layer and finally a classification head, giving predictions for each input token. However, in the proposed system, there is a new layer inserted just after the XLM-RoBERTa embeddings calculation and before the linear layer. The resulting system architecture is presented in Figure 1.

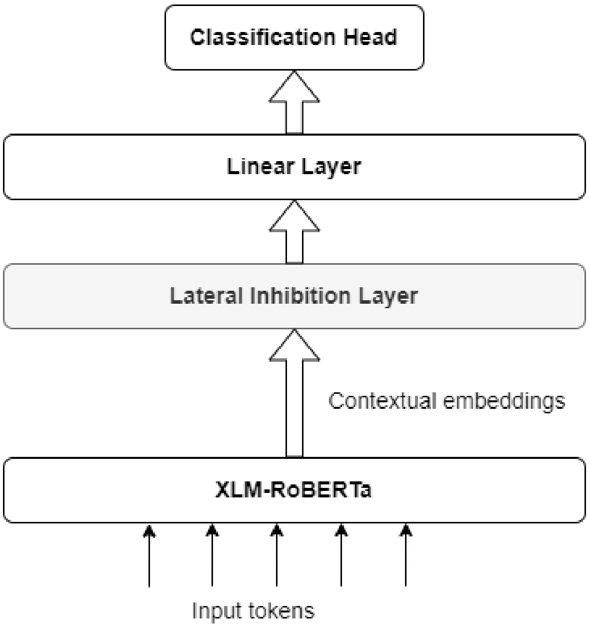

Figure 1: System architecture

The newly introduced layer follows the biological process of lateral inhibition. Thus, an embedding value is either allowed to pass unchanged to the next layer or set to zero, depending on the other values. Similar to a linear layer, a matrix of weights (W) and a bias (B) were kept. However, the diagonal values of the weights matrix were always set to zero to allow only interaction from adjacent neurons. Furthermore, the Heaviside function (see Equation 1) was applied to determine which values pass through the layer or become zero. The equation associated with the forward pass is given in Equation 2, where X is the layer's input vector, associated with a token embedding representation, $Diag$ represents the matrix diagonal, $ZeroDiag$ is the matrix with the value zero on the diagonal, and W and B represent the weights and bias.

$$\Theta(x) = \begin{cases} 1, x > 0 \\ 0, x \leq 0 \end{cases} \tag{1}$$

$$F(X) = X * Diag(\Theta(X * ZeroDiag(W) + B)) \tag{2}$$

To overcome the problem of computing a derivative for the Heaviside function, in the backwards pass the Heaviside function was approximated with the sigmoid function with a scaling parameter (Wunderlich and Pehle, 2021). This is described in Equation 3. The derivative of the sigmoid function is given in Equation 4, where $\sigma(x)$ is the same as in Equation 3. This approximation technique is also

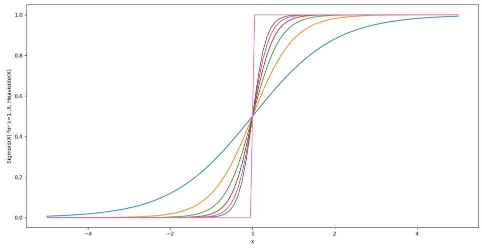

Figure 2: $\sigma(x)$ for k=1..6 and $\Theta(x)$

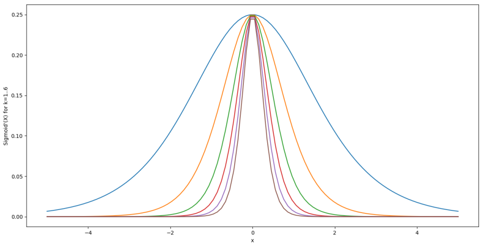

Figure 3: $\sigma'(x)$ for $k = 1..6$

known as surrogate gradient learning (Neftci et al., 2019). It allows the use of a non-differentiable function in the forward pass (in this case the Heaviside function) and approximates the derivative in the backwards pass by means of a different function.

$$\sigma(x) = \frac{1}{1 + e^{-kx}} \tag{3}$$

$$\sigma'(x) = k\sigma(x)\sigma(-x) \tag{4}$$

Figure 2 shows a plot for the Sigmoid function, for various values of the scaling parameter k, superimposed on a plot of the Heaviside function. Higher values for the scaling parameter give better approximations for the Heaviside function. A plot for the derivative of the Sigmoid function with $k = 1..6$ is given in Figure 3.

The entire system was then implemented in PyTorch following the system diagram given in Figure 1. The embedding vector associated with each token is processed by the lateral inhibition layer, then goes through a linear layer using ReLU as the activation function and finally through the classification head.

4.2 Experiments

Training started with the creation of a multilingual model based on the provided multilingual corpus. Training was performed for 40 epochs, with the best performing model on the dev dataset being

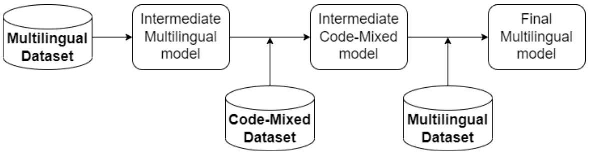

Figure 4: Training flow for creating the multilingual model

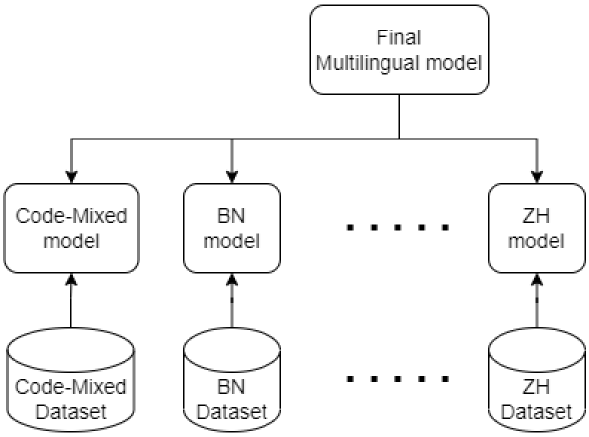

Figure 5: Language-specific training

stored. The resulting intermediate model was fine-tuned with the code mixed data, for another 40 epochs. Thus, all the available data were now included in the new intermediate model. This model was then used as the basis for further training on the multilingual dataset (another 40 epochs), producing the final multilingual model. The results from this final model were submitted in the multilingual task. This training procedure is described in Figure 4.

The final multilingual model was then used as a starting point for the other models. The architecture was the same and the model parameters were initialized with the multilingual parameters. Fine-tuning was performed for 80 epochs for all languages and in the case of the code-mixed dataset. The best performing model for each task was stored and the results were submitted to the corresponding category. The language-specific training procedure is described in Figure 5.

The models were trained using a single Nvidia Quadro RTX 5000 GPU board. The dataset was first pre-processed, as described in Section 3. Several experiments were performed to determine the best parameters. However, due to limited time and hardware resources, it was not possible to perform an exhaustive search. The participating models employed learning rates of $1e - 05$ (the majority of the models) or $2e - 05$ (Bangla and code-mixed).

Track	With Lateral Inhibition				Without Lateral Inhibition				Diff
	P	**R**	**F1**	**Epoch**	**P**	**R**	**F1**	**Epoch**	**F1**
Bangla	**68.08**	**65.33**	**66.28**	40	65.95	63.99	64.68	68	1.60
Chinese	**68.17**	62.05	62.70	35	67.05	**64.39**	**64.41**	56	-1.71
Dutch	78.82	**78.30**	**78.41**	63	**78.85**	77.27	77.25	37	1.16
English	76.54	75.35	75.78	66	**76.59**	**76.09**	**76.21**	53	-0.43
Farsi	**70.77**	**70.45**	**70.42**	36	70.10	68.84	69.24	21	1.18
German	80.01	78.97	79.39	37	**80.19**	**79.44**	**79.70**	71	-0.31
Hindi	69.05	67.77	68.08	42	**70.14**	**68.18**	**68.87**	65	-0.79
Korean	**72.06**	71.93	71.74	8	71.83	**72.58**	**71.99**	70	-0.16
Russian	75.86	73.83	74.60	62	**75.89**	**73.94**	**74.68**	62	-0.08
Spanish	**76.21**	**75.43**	**75.62**	4	75.88	75.25	75.36	58	0.26
Turkish	**71.81**	**70.25**	**70.42**	78	71.28	69.14	69.70	74	0.72
Multilingual	**72.25**	**72.78**	**72.10**	33	71.78	72.72	71.87	17	0.23
Code-mixed	**79.58**	**79.23**	**79.37**	63	78.93	79.02	78.95	62	0.42
Intermediate Multilingual	**71.53**	**72.34**	**71.50**	34	70.48	71.11	70.35	14	1.15
Intermediate Code-mixed	**79.28**	**79.42**	**79.31**	31	78.50	79.00	78.73	28	0.58

Table 1: Results on the test dataset for all tracks.

A dropout value of 0.1 was used for both the lateral inhibition and linear layers. For the backwards pass approximation of the Heaviside function, the sigmoid scaling parameter was set to 10.

5 Results

The models produced by the system described in the previous sections participated in all the MultiCoNER tracks. The official ranking metric was macro-averaged F1. The results are given in Table 1 for all tracks, in alphabetical order of the languages, while the multilingual and code-mixed tracks are shown at the end of the table.

The best best average F1 score was achieved in the German track (79.39%), followed closely by the code-mixed track (79.37%). All these results were obtained by using only the provided dataset, without external resources. For most of the tracks precision and recall are similar (with precision being approximately 1% higher than recall), with the exception of Chinese (precision is 6% higher than recall) and Russian (precision is 2% higher than recall).

In Table 1 is also given a comparison between the results obtained using the new lateral inhibition layer and the results obtained by the same system without the lateral inhibition layer. In 7 tracks (Bangla, Dutch, Farsi, Spanish, Turkish, Multilingual, Code-mixed) the new layer improved the overall F1 score, in 1 track (Russian) the results were roughly the same (a difference of 0.08 %), while in 5 tracks (Chinese, English, German, Hindi, Korean) the new layer actually decreased the system's performance. By looking at the size of the training corpora used in XLM-RoBERTa, as reported by Conneau et al. (2020), it seems the new layer improved the system performance in the case of languages represented by less than 54Gb of data. This seems to confirm the intuition behind the new layer, that the model is able to better focus on details present in the data and potentially filter out the noise. There are two exceptions: Farsi, trained on 111 Gb of data, and Hindi, trained on 20 Gb. In this case additional investigation should probably be performed with regard to the quality of the data used for training the model, possibly taking into account the relative language complexity (Bentz et al., 2016).

Table 1 also presents the training epochs associated with the best model. For some languages, the new layer is able to reduce the number of training epochs, but there is no clear pattern for when this happens. Interestingly however, for Spanish and Korean the new layer is able to reduce the training time to less than 10 epochs.

In the multilingual and code-mixed tracks, the model enhanced with the lateral inhibition layer

Track	LOC	PER	PROD	GRP	CW	CORP
Bangla	69.03	77.41	62.92	73.46	49.57	65.31
Chinese	72.30	64.71	67.93	45.44	58.53	67.27
Dutch	80.85	89.82	76.99	75.14	71.35	76.29
English	77.29	90.11	73.27	72.93	66.76	74.29
Farsi	74.04	79.19	70.71	72.33	58.59	67.67
German	82.19	90.24	78.66	75.28	73.39	76.57
Hindi	70.93	75.14	66.35	68.70	57.13	70.25
Korean	76.54	77.86	72.48	68.02	64.02	71.51
Russian	74.05	80.30	75.04	71.29	70.09	76.84
Spanish	76.95	89.27	70.81	71.61	68.69	76.40
Turkish	72.44	81.80	73.01	64.85	61.91	68.49
Multilingual	76.84	83.85	68.82	64.89	66.72	71.47
Code-mixed	82.29	88.29	81.09	73.13	73.99	77.42

Table 2: F1 scores on the test dataset for all tracks and for all entity types.

was able to improve both precision (by 0.47% for multilingual and 0.65% for code-mixed) and recall (by 0.06% for multilingual and 0.21% for code-mixed), leading to corresponding F1 differences. Nevertheless, the number of epochs is not reduced for these tracks. As described in Section 4.2 and illustrated in Figure 4, there were also two intermediate models (one multilingual and one code-mixed). The results for these models are also provided in Table 1. The proposed approach increased the multilingual F1 score by 0.6% for the lateral inhibition system and by 1.52% without lateral inhibition. Similarly, the code-mixed performance was increased by 0.06% (with lateral inhibition) and by 0.22% (without lateral inhibition). It seems in this case that the presence of the lateral inhibition layer reduced the overall gain obtained from training the model in multiple stages. Nevertheless, the multi-stage approach increased the final F1 score.

Table 2 presents the F1 results obtained with the lateral inhibition layer, for each track and for each entity type. It can be noticed that commonly used named entities (locations and persons) achieve the highest score in all tracks. The hardest to predict (achieving the lowest scores) are group (GRP) and creative work (CW). This observation holds for all tracks, but particularly in Chinese, the group entity obtains the lowest score (45.44%), and in Bangla, the creative works entity obtains 49.57%. It seems that these low values are due to the nature of these entities, their complexity and their evolution over time. Furthermore, by looking at the dataset structure for Chinese, the group entity type was the least represented in the training set. Moreover, by looking at the number of unique entities present in the dataset, there is a high discrepancy between unique training entities, for group and creative works, and the corresponding unique instances in the test set, especially for Chinese and Bangla.

6 Conclusion

The system described in this paper participated in the MultiCoNER shared task. It made use of a new artificial neural network layer inspired by the biological process of lateral inhibition. By means of this mechanism, the system achieved third place in the general ranking associated with 7 languages (Spanish, Dutch, Russian, Korean, Farsi, German, and Hindi). There were no additional resources employed, apart from the provided dataset (no additional text and no gazetteers). A simple data augmentation method (as described in Section 3) was applied only for the code-mixed track, while still using only the provided data.

Experiments performed after the task deadline, showed (as reported in Section 5) that not all tracks benefit from using the new layer. It is likely that languages with an increased number of tokens present when training the multilingual XLM-RoBERTa model will benefit less from employing the new layer. Nevertheless, models with the lateral inhibition layer trained for lower resourced languages, multilingual, and code-mixed datasets seem to achieve higher scores. Furthermore, improvements can be seen in both precision and recall.

The proposed lateral inhibition layer can be applied to other natural language processing tasks as well. In the future more experiments will be conducted with this layer to determine its suitability in other contexts.

References

Gustavo Aguilar, Suraj Maharjan, Adrian Pastor López-Monroy, and Thamar Solorio. 2017. A multi-task approach for named entity recognition in social media data. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 148–153, Copenhagen, Denmark. Association for Computational Linguistics.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Sandeep Ashwini and Jinho D. Choi. 2014. Targetable named entity recognition in social media. *CoRR*, abs/1408.0782.

Christian Bentz, Tatyana Ruzsics, Alexander Koplenig, and Tanja Samardžić. 2016. A comparison between morphological complexity measures: Typological data vs. language corpora. In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 142–153, Osaka, Japan. The COLING 2016 Organizing Committee.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

Ronald A. Cohen. 2011. *Lateral Inhibition*, pages 1436–1437. Springer New York, New York, NY.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Franck Dernoncourt, Ji Young Lee, and Peter Szolovits. 2017. NeuroNER: an easy-to-use program for named-entity recognition based on neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 97–102, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Besnik Fetahu, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2021. Gazetteer Enhanced Named Entity Recognition for Code-Mixed Web Queries. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1677–1681.

Ingo Glaser, Bernhard Waltl, and Florian Matthes. 2018. Named entity recognition, extraction, and linking in german legal contracts. In *IRIS: Internationales Rechtsinformatik Symposium*, pages 325–334.

Aitor Gonzalez-Agirre, Montserrat Marimon, Ander Intxaurrondo, Obdulia Rabal, Marta Villegas, and Martin Krallinger. 2019. PharmaCoNER: Pharmacological substances, compounds and proteins named entity recognition track. In *Proceedings of The 5th Workshop on BioNLP Open Shared Tasks*, pages 1–10, Hong Kong, China. Association for Computational Linguistics.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Yuting Hu and Suzan Verberne. 2020. Named entity recognition for Chinese biomedical patents. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 627–637, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Radu Ion, Vasile Păiș, and Maria Mitrofan. 2019. RACAI's system at PharmaCoNER 2019. In *Proceedings of The 5th Workshop on BioNLP Open Shared Tasks*, pages 90–99, Hong Kong, China. Association for Computational Linguistics.

Jörg Landthaler, Bernhard Waltl, and Florian Matthes. 2016. Unveiling references in legal texts-implicit versus explicit network structures. In *IRIS: Internationales Rechtsinformatik Symposium*, volume 8, pages 71–8.

Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. 2019. Fine-grained named entity recognition in legal documents. In *Semantic Systems. The Power of AI and Knowledge Graphs*, pages 272–287, Cham. Springer International Publishing.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. Multiconer: a large-scale multilingual dataset for complex named entity recognition.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. Semeval-2022 task 11: Multilingual complex named entity recognition (multiconer). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Tao Meng, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2021. GEMNET: Effective gated gazetteer representations for recognizing complex entities in low-context input. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1499–1512, Online. Association for Computational Linguistics.

Antonio Miranda-Escalada, Eulàlia Farré-Maduell, Salvador Lima-López, Luis Gascó, Vicent Briva-Iglesias, Marvin Agüero-Torales, and Martin Krallinger. 2021. The ProfNER shared task on automatic recognition of occupation mentions in social media: systems, evaluation, guidelines, embeddings and corpora. In *Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task*, pages 13–20, Mexico City, Mexico. Association for Computational Linguistics.

Emre O. Neftci, Hesham Mostafa, and Friedemann Zenke. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Vasile Păiș and Maria Mitrofan. 2021. Assessing multiple word embeddings for named entity recognition of professions and occupations in health-related social media. In *Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task*, pages 128–130, Mexico City, Mexico. Association for Computational Linguistics.

Vasile Păiș, Maria Mitrofan, Carol Luca Gasan, Vlad Coneschi, and Alexandru Ianov. 2021. Named entity recognition in the Romanian legal domain. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 9–18, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Burr Settles. 2004. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*, pages 107–110, Geneva, Switzerland. COLING.

Jana Straková, Milan Straka, and Jan Hajic. 2019. Neural architectures for nested NER through linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331, Florence, Italy. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2020. More embeddings, better sequence labelers? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3992–4006, Online. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021. Automated concatenation of embeddings for structured prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2643–2660, Online. Association for Computational Linguistics.

Timo C. Wunderlich and Christian Pehle. 2021. Event-based backpropagation can compute exact gradients for spiking neural networks. *Scientific Reports*, 11(1):12829.

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.