Improved Biomedical Entity Recognition via longer context modeling

Nikolaos Stylianou[1], Panagiotis Kosmoliaptsis[1,2], and Ioannis Vlahavas[1]

[1] Aristotle University of Thessaloniki, Thessaloniki 54124, Greece
[2] General Hospital of Thessaloniki "George Papanikolaou", Thessaloniki 57010, Greece
{nstylia,kosmoliap,vlahavas}@csd.auth.gr

Abstract. Biomedical Named Entity Recognition is a difficult task, aimed to identify all named entities in medical literature. The importance of the task becomes apparent as these entities are used to identify key features, enable better search results and can accelerate the process of reviewing related evidence to a medical case. This practice is known as Evidence-Based Medicine (EBM) and is globally used by medical practitioners who do not have the time to read all the latest developments in their respective fields. In this paper we propose a methodology which achieves state-of-the-art results in a plethora of Biomedical Named Entity Recognition datasets, with a lightweight approach that requires minimal training. Our model is end-to-end and capable of efficiently modeling significantly longer sequences than previous models, benefiting from inter-sentence dependencies.

Keywords: Natural Language Processing · Deep Learning · Biomedical Named Entity Recognition · Evidence Based Medicine.

1 Introduction

Named Entity Recognition (NER) refers to the Natural Language Processing (NLP) task of identifying all the Named Entities in a span of text. It can be used as an Information Extraction (IE) tool, to identify important entities in texts, but also finds uses in many downstream tasks such as question answering, summarization, information retrieval and knowledge graph construction [20]. When the source information is from documents in the clinical domain, this task is separately identified as Biomedical Named Entity Recognition (BioNER) and the focal point is to identify entities of interest in that domain.

The entities that are recognized by BioNER systems are focused on either Disease, Chemicals, Genes, Molecules of Cells and Drugs, or a combination of the above. As a result, BioNER's domain of application extends that of classic NER systems to domain specific tasks such as, adverse drug event extraction [10], drug-drug interactions [35] and protein-protein interactions [27].

Daily, a staggering number of new research is published in the plethora of biomedical fields, with the number of articles of interest to a medical practitioner

increasing exponentially. Evidence-Based Medicine (EBM) is the practice with which medical practitioners identify all the relevant previous research, usually in the form of Clinical Trials (CTs) or Randomized Control Trials (RCTs), to create informed treatment plans. The most dominant method to achieve this is through the PICO Framework, named after its elements Population, Intervention, Comparator and Outcome [21].

A combination of advancements in both Deep Learning (DL) and Natural Language Processing (NLP) techniques has contributed significantly in the increased performance of modern BioNER systems [11,5,33]. In comparison to general purpose Named Entity Recognition (NER), BioNER systems suffer in performance due to the high variance and complexity of terms found in medical literature [3]. This complexity is worsened in the case of EBM, in which a term, e.g. "high blood sugar", can be identified in multiple classes, i.e. Population or Outcome in this case, depending on the context of the study.

Current state-of-the-art approaches across all entity domains, attempt to identify the entities described in a sequence, in a sentence by sentence manner. With such an approach, inter-sentence dependencies are never considered. Consequently, in cases such as EBM, in which entities can change their class depending on the document's context, considering such dependencies can be the determining factor between for example identifying "high blood sugar" as a Population rather than an Outcome or vice-versa.

In this work, we propose a state-of-the-art, transformer based, BioNER model. Our approach is based on Transformers, requires minimal training as it makes use of transfer learning techniques and can effectively model significantly longer sequences than traditional models. Effectively, we are able to model abstracts instead of sentences in one pass, utilizing long term dependencies during both training and inference. We evaluate our model in both Disease, Chemicals, Molecules of Cells (cell lines, genes and proteins) and EBM data, where we showcase an overall performance increase compared to the previous state-of-the-art.

2 Related Work

Biomedical Named Entity Recognition has been of significant research interest due to its distinctive applications in a plethora of scientific fields [20]. For that reason, there is an abundance of research works and datasets that refer to different field of bio-medicine [12]. Two major categories of interest have been identified in BioNER, for the task of Named Entity Recognition, identifying a specific entity type [11,5], and identifying all specified entity types using multiple datasets [13,33].

The first category is more limited in scope, with their contributions mainly focused on architectural changes that improve the performance when identifying a specific entity category. A significant number of research has been completed in such models [12], with the most recent models attempting to tackle the distinct issue in BioNER of longer entity sequences. In [11], they propose a model architecture to create a combination of unary and pairwise hidden states to

increase the models expressiveness over longer entity sequences. A different approach introduces a combinatorial embedding [5], making use of two character level architectures and a word level architecture to create token representations. The expressiveness of Recurrent Neural Networks (RNNs), and more specifically Bidirectional Long Short-term Memory (BiLSTM) networks for the main model architecture has been the key component to both studies.

The focus of the second category has been on developing a more general BioNER system which is not bound to one entity type. Due to the plurality of entities and their different scopes, these entities are only identified to their highest level, i.e. Disease, Drugs, Chemical Compounds. In order to effective identify all entities, an ensemble of entity specific models based on BiLSTMs is introduced in [32]. The model is later extended with the use of Flair embeddings [33], to further increase its performance. Levering the pre-trained BioBERT model to extract more informed word representations, a multi-task learning approach is introduced in [13], with entity specific layers to handle multiple entity predictions. A significant flaw to such approaches is that while the test set is a union of the respective single entity dataset test sets, each sequence still only identifies one entity, making it easier for the respective models to build dependencies.

Evidence-Based Medicine approaches, much like the second category attempt to identify more than one entity type in each sequence. However, instead of generic bio-medicine terms, the entities are identified using the PICO Framework, and as such approached with a variety of ways. The best performing NER EBM system [26], is introducing a deep BiLSTM-based model with residual connections to model the intersequence dependencies. The current state-of-the-art model [25] is solving EBM as a Question-Answering task, using the whole documents as context to extract the PICO entities.

3 LongSeq: Our proposed approach

LongSeq is an end-to-end Transformer-based model that enables the effective modeling of longer sequences in each step, resulting in better overall performance. In comparison to traditional RNN based approaches, LongSeq does not suffer from vanishing gradient when modeling very long sequences [23], due to the Transformer-based architecture.

Furthermore, our approach is faster to train overall, requiring both less time per iteration as well as less training epochs than traditional approaches. This is achieved with a combination of a pre-trained language model for word representations as well as computationally efficient transformer architectures that can model such long sequences. As a result, it can model dependencies past the sentence scope of previous approaches, which can be efficiently used by Conditional Random Fields (CRF) during prediction [18].

3.1 Our model

Our proposed model, is an end-to-end Transformer-based model consisting of a pre-trained BERT model to create contextualized word representations, stacked

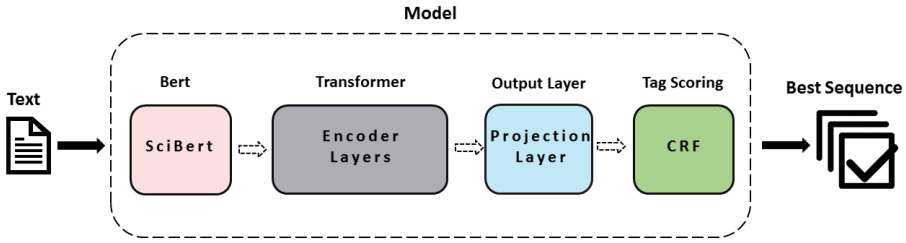

Fig. 1. LongSeq model architecture

Transformer encoders and a CRF layer to handle predictions. Fig. 1 illustrates the architecture of LongSeq.

Our model takes as input a sequence of tokens $W = (w_1, \ldots, w_l)$, where l is the max number of tokens in each input. This sequence is transformed into an embeddings sequence $X = (x_1, \ldots, x_n)$ after being passed through a BERT model, such that $x_i \in \mathbb{R}^{n \times d_{BERT}}$, where n is the number of vectors in the final sequence after preprocessing and padding and d_{BERT} is the BERT embeddings size. Each vector in X is forwarded through N stacked Transformer encoder layers, each with h number of attention heads, in order to create better representations of the input sequence. The resulting vector is of size $X \in \mathbb{R}^{n \times d_{Enc}}$, with d_{Enc} being the encoder's hidden size, which will be passed through a projection layer for dimensionality reduction to $X \in \mathbb{R}^{n \times d_{Proj}}$, followed by a CRF layer that handles predictions. The CRF uses the Viterbi algorithm to efficiently predict the most likely sequence of labels [18].

3.2 Transformer Encoders

LongSeq is modularly designed to use a number of different Transformer architectures. In our approach, we experiment with four different Transformer encoder architectures designed for computational efficiency via changes to their attention mechanisms.

A Transformer can represent both an Encoder and a Decoder layer, with some architectural differences. However, in our approach we are only interested in the Encoder component. Each Transformer encoder has two sub-layers blocks, each wrapped with a normalization layer and a residual connection around them. The first block consists of a multi-headed self attention mechanism and the second consists of a position-wise fully connected feed-forward layer. Each layer takes as input a vector $X \in \mathbb{R}^{n \times d_{Enc}}$, where n is the sequence length and the d_{Enc} is the model specific hidden size, and outputs a similar sized vector. Formalized, the Transformer encoder can be expressed as:

$$E_A = \texttt{LayerNorm}(\texttt{MultiHeadSelfAttention}(X)) + X \tag{1}$$

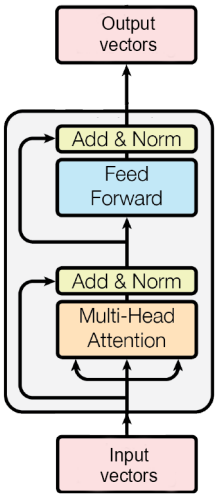

Fig. 2. Transformer Encoder layer [29]

$$E_B = \texttt{LayerNorm}(\texttt{PositionFFN}(E_A)) + E_A \tag{2}$$

where X is the input vector and E_B will become the input (X) in the following Transformer layer. Detailed architectural information are provided in [29].

However, the Multi-headed self attention mechanism proposed in [29] is inefficient, leading to quadratic computational complexity – $O(n^2)$. This is attributed to the dense operation in each attention head in which each element in the sequence learns to gather from the other tokens in the sequence. As a result, the bigger the input sequence, the more computationally demanding is the model.

In our model, except from Vanilla Transformers, we consider three alternative Transformer architectures, which change the attention mechanism to introduce different forms of sparsity in order to reduce the computational complexity.

Reformer [15] is based on locality sensitive hashing (LSH). Consequently, the query-keys are hashed into buckets using a random projection. The intuition behind the LSH is that nearby vectors should maintain similar hash, and hence require less computations leading to a computational complexity of $O(n\ log(n))$. In a different approach, Longformer [2] introduces a combination of local sliding windowed attention and task-specific global attention. The introduced sparsity in the attention mechanism lowers the original quadratic computation complexity to $O(n \times w))$, where w is the window size. The last architecture we experimented with, is Linformer [30], which uses a low rank projection technique on the length dimension to project the dimensional keys and values in the Query, Key, Value attention scheme to a lower dimension (k). As a result, the computational complexity of Linformer is $O(n)$ due to k being sufficiently small and

stable. A detailed survey on the efficiency of different Transformer architectures, including the aforementioned, has been conducted in [28].

4 Experiments

To evaluate the effectiveness of our approach, we compare the performance of our model on four datasets, with four different scopes (EBM, Diseases, Molecules of Cells, Chemicals). The models are tested while modeling sentence-level input as well as whole abstracts.

4.1 Data and Processing

In using four benchmark datasets, we attempt to cover the majority of biomedical fields. As shown in Table 1, in this study we use EBM-NLP [22], NCBI-Disease [8], JNLPBA [14] and SCAI-Chemicals [17], covering four different biomedical entity categories. All of the aforementioned datasets are from abstracts of publications from PubMed/MEDLINE. With the exception of EBM-NLP, which poses the restriction that the abstracts must come from CTs or RCTs type studies, the other datasets are collected using MeSH terms to limit the search results.

Table 1. Dataset characteristics

Dataset	Entities	# of Sents	# of Absts	Max words per Sent	Max words per Abst	% over 512 tokens
EBM-NLP	PICO	53397	4982	241	838	1.69%
NCBI-Disease	Disease	7421	793	124	501	0%
JNLPBA	Gene & Proteins	24716	2404	208	645	0.41%
SCAI-Chemicals	Chemicals	965	100	169	666	8.08%

Sent(s) is used to identify Sentence(s) and Abst(s) is used to identify Abstract(s).

EBM-NLP is annotated under the PICO scheme, identifying *Population*, *Intervention/Comparator* and *Outcome* as the three main classes. Its inherited difficulty stems from the possible change in entity class of a span of text, depending on the context. Similarly, NCBI-Disease is used to evaluate the performance of our approach on identifying *Disease* class entities in medical documents. Both of these datasets annotate entities on a higher level, which contain specific subclasses for their entity types. SCAI-Chemical identifies eight classes of Chemicals (*IUPAC*, *PART*, *TRIVIAL*, *ABB*, *SUM*, *FAMILY*, *MODIFIER* and *TRIVIAL-VAR*). Finally JNLPBA is consisted of five classes of Molecules of Cells (*DNA*, *RNA*, *CellLine*, *CellType*, *Protein*).

In all datasets we use, Inside-Outside (IO) annotations scheme, as we are not dealing with nested entities. Furthermore, we tokenize all documents using BERT's Byte-Pair Encoding (BPE) tokenizer, and adding the special CLS and SEP tokens appropriately. Lastly, we apply padding to 512 tokens when necessary.

4.2 Experimental Setup

We train LongSeq on all datasets previously described, taking as input either sentences ($LongSeq_{sent}$) or whole abstracts ($LongSeq_{abst}$). We are also using different transformer encoders, each with its own set of parameters, depending on the dataset.

In all models we experimented with different batch sizes {6,8,16,32}, learning rates {3e-3,3e-4,3e-5} and training epochs {2,4,6,8,10}. SciBERT [1] was used for word embeddings, with $d_{BERT} = 768$. Our final models are trained for 6 epochs with a learning rate of 3e-5. Having two input scopes, we used different batch sizes to consider the same amount of context per training step, with $LongSeq_{sent}$ having a batch size of 32 and $LongSeq_{abst}$ having a batch size of 4.

For all transformer encoders we considered combinations of {2,4,6,8} number of layers (N) and {2,4,6,8} number of heads (h), regardless of input scope. In our final models, Vanilla Transformer and Reformer both have $N = 6$ and $h = 6$. Longformer [2] has $N = 4$, $h = 4$ and uses the sliding chunks attention approach with a window size $w = 256$. Finally, Linformer [30] is defined with $N = 4$, $h = 4$ and $k = 512$ where k is a special parameter to define the projection length inside the sparse attention mechanism. In all models we define a maximum sequence length of 512 tokens, $d_{Enc} = 768$ and d_{Proj} to the number of entity classes.

All experiments were run using Google Colab, with a P100 GPU (16GB vRAM). The code to reproduce our experiments is available on GitHub[3].

4.3 Results

We compare the performance of our best LongSeq model, with the state-of-the-art approaches in all four datasets in terms of macro-averaged overall Precision, Recall and F1-score, following past approaches to enable direct comparisons.

In all cases we use the reported scores from the respective publications without reproducing the experiments. We opt to use the reported scores as we are comparing LongSeq to a total of 15 models, the majority of which are only tested on one of the datasets and designed for that specific use-case.

Table 2. Results on EBM-NLP dataset

Models	Precision	Recall	F1
LongSeq	**79**%	**81**%	**79**%
EBM+ [26]	70%	70%	71%
BERT [7]	69%	66%	68%
EBM-PICO [22]	71%	64%	67%
QA-PICO [25]	-	-	<u>75</u>%

Table 3. Results on NCBI-Disease dataset

Models	Precision	Recall	F1
LongSeq	**96**%	**96**%	**96**%
BioBert [19]	<u>89</u>%	<u>89</u>%	<u>89</u>%
MT-BioNER [13]	86%	89%	88%
CNN-BiLSTM [5]	86%	87%	86%
Gramm-Cnn [4]	84%	84%	84%
CollaboNet[34]	83%	85%	84%
Spark [16]	-	-	<u>89</u>%

[3] https://github.com/AUTH-MINT/LongSeq/

In Table 2 we present the comparative performance of our best model with recent research on EBM-NLP dataset. We achieve a 4% overall increase when identifying Population, Intervention/Comparator and Outcomes, with a 10% increase from the second best approach.

When compared to other models on NCBI-Disease (Table 3), LongSeq achieves a 7% macro-average F1 increase, compared to Spark [16] and 7% increase in all measures compared to BioBERT [19].

We follow a similar trend with a 20% overall increase when comparing the performance of LongSeq on identifying Chemical entities in Table 5. Most notably, other approaches that test on SCAI-Chemicals dataset have big variance in both Precision and Recall, leading to lackluster F1 scores. Our approach is consistent across all metrics.

Unavailingly, LongSeq performs worse from the compared models on JNLPBA dataset (Table 4). While we manage to get a high average Recall, our model struggles in terms of Precision leading to an insignificant outcome. Compared to the other approaches which use character level features to increase their Precision, LongSeq considers inputs at a token level, as per SciBERT.

We separate the scores of QA-PICO and Spark from the other implementations, in Tables 2 and 3 & 4 respectively, as the original publications do not report detailed results. We use **bold** to identify the best performance per metric and we also underline the scores of the second-best model.

Table 4. Results on JNLPBA dataset

Models	Precision	Recall	F1
CollaboNet[34]	**72**%	**82**%	**77**%
Gridach[9]	74%	77%	76%
MTM-CW [31]	70%	76%	73%
MO-MTM [6]	-	-	69%
LongSeq	67%	80%	65%
Spark [16]	-	-	**81**%

Table 5. Results on SCAI-Chemicals dataset

Models	Precision	Recall	F1
LongSeq	**91**%	**87**%	**88**%
ChemSpot [24]	67%	69%	68%
OSCAR3 [17]	52%	72%	60%
ChemSpot+CRF [24]	88%	28%	42%

4.4 Ablation study

LongSeq's performance depends on both the input scope and the Transformer architecture. We performed a comparative study, in both scopes with all Transformer architectures to discover the best setup per biomedical domain. We repeated each experiment five times to account for any fluctuation in the results. In Tables 6, 7, 8 & 9 we use **bold** for the best performing architecture in abstract level and underline for the best model in sentence level.

At sentence level, there is no clear better architecture, as all approaches tend to have at least one best performance for one biomedical domain. At abstract

Table 6. Architecture and scope results on EBM-NLP Dataset

Transformer Architectures	Sentences			Abstracts		
	P	R	F1	P	R	F1
Vanilla	76%	77%	75%	79%	80%	78%
Linformer	75%	79%	76%	81%	78%	78%
Reformer	78%	79%	78%	79%	80%	78%
Longformer	75%	77%	75%	**79%**	**81%**	**79%**

Table 7. Architecture and scope results on NCBI-Disease Dataset

Transformer Architectures	Sentences			Abstracts		
	P	R	F1	P	R	F1
Vanilla	95%	96%	95%	96%	95%	96%
Linformer	96%	95%	95%	95%	96%	96%
Reformer	45%	50%	47%	45%	50%	47%
Longformer	95%	95%	95%	**96%**	**96%**	**96%**

level, Longformer performs best in all datasets, with the exception of JNLPBA, which is also our worst performing one.

It is noteworthy to mention that the best performing architecture per dataset is usually different when changing the input scope from sentences to abstracts. This is on par with our initial expectations based on the different attention mechanisms used in the selected architectures.

Table 8. Architecture and scope results on JLNBPA Dataset

Transformer Architectures	Sentences			Abstracts		
	P	R	F1	P	R	F1
Vanilla	63%	74%	57%	**67%**	**80%**	**65%**
Linformer	63%	77%	59%	64%	79%	63%
Reformer	62%	76%	58%	**67%**	**80%**	**65%**
Longformer	62%	75%	57%	66%	79%	63%

Table 9. Architecture and scope results on SCAI-Chemical Dataset

Transformer Architectures	Sentences			Abstracts		
	P	R	F1	P	R	F1
Vanilla	83%	78%	75%	94%	78%	84%
Linformer	1%	11%	0%	90%	86%	87%
Reformer	85%	72%	76%	**91%**	**87%**	**88%**
Longformer	85%	72%	76%	**91%**	**87%**	**88%**

Finally, we compare the overall training times of Longformer, our best performing architecture overall, in all datasets and in both scopes in Table 10.

Table 10. Longformer training time comparison

Dataset	Sentences	Abstracts
EBM-NLP	722m 38s	93m 19s
NCBI	111m 40s	15m 52s
JNLPBA	455m 13s	49m 47s
SCAI-Chemical	24m 47s	3m 2s

From Table 10, it becomes obvious that models that use abstracts need much less time for training due to smaller number of instances (Table 1 – 3rd & 4th columns) in otherwise using sentences and due to less required backward optimization steps.

5 Discussion

In this study we presented a novel NER architecture, with stacked Transformers, for BioNER. Our approach can be trained on longer sequences that previous approaches due to the novel Transformer architecture and does not suffer from vanishing gradient. We showcased state-of-the-art performance in three of the four biomedical domains that we tested our approach, boasting an average 10% overall increase for Disease, EBM and Chemicals.

Our model makes used of SciBERT, a pre-trained BERT language model on scientific documents. This leads to word representations and requires less training time. We also compare our architecture with four different Transformer architectures, the Vanilla Transformer and three other approaches designed for computational efficiency by introducing attention sparsity.

We experimentally prove that Longformer is more suitable for the majority of cases, while Reformer and Vanilla, are better suited when predicting Molecules of Cells. Furthermore, we show a degrade performance in Molecules of Cells compared to other architectures which is attributed to lacking word representations for this task and absence of character level features.

LongSeq utilizes the whole abstract instead of each individual sentence as an input, leading to better performance and faster training times. This is due to the fact that in sentence level the number of predictions required is increased due to sentence padding at batch level. This increased number of predictions and error propagation cycles is computationally expensive and time consuming. Moreover, our approach instills more context to each sequence leading to better predictions.

6 Conclusions

In this paper we introduced LongSeq, a long context BioNER model. LongSeq leverages the contextual information of the whole abstract, which along with a combination of stacked Transformers results in state-of-the-art performance in EBM, Disease and Chemical identification. We also experimentally evaluate the performance of different Transformer architectures in the task, emerging Longformer as the best option concerning overall performance.

In the future, we aim to expand our current work to consider both word and character level features to increase generalizability and cover more medical domains of applications, creating a holistic approach to medical information extraction.

References

1. Beltagy, I., Lo, K., Cohan, A.: Scibert: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676 (2019)
2. Beltagy, I., Peters, M.E., Cohan, A.: Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150 (2020)

3. Campos, D., Matos, S., Oliveira, J.L.: Biomedical named entity recognition: a survey of machine-learning tools. Theory and Applications for Advanced Text Mining **11**, 175–195 (2012)
4. Cho, H., Lee, H.: Biomedical named entity recognition using deep neural networks with contextual information. BMC bioinformatics **20**(1), 1–11 (2019)
5. Cho, M., Ha, J., Park, C., Park, S.: Combinatorial feature embedding based on cnn and lstm for biomedical named entity recognition. Journal of biomedical informatics **103**, 103381 (2020)
6. Crichton, G., Pyysalo, S., Chiu, B., Korhonen, A.: A neural network multi-task learning approach to biomedical named entity recognition. BMC bioinformatics **18**(1), 1–14 (2017)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
8. Doğan, R.I., Leaman, R., Lu, Z.: Ncbi disease corpus: a resource for disease name recognition and concept normalization. Journal of biomedical informatics **47**, 1–10 (2014)
9. Gridach, M.: Character-level neural network for biomedical named entity recognition. Journal of biomedical informatics **70**, 85–91 (2017)
10. Gurulingappa, H., Mateen-Rajpu, A., Toldo, L.: Extraction of potential adverse drug events from medical case reports. Journal of biomedical semantics **3**(1), 1–10 (2012)
11. Hong, S., Lee, J.G.: Dtranner: biomedical named entity recognition with deep learning-based label-label transition model. BMC bioinformatics **21**(1), 53 (2020)
12. Huang, M.S., Lai, P.T., Lin, P.Y., You, Y.T., Tsai, R.T.H., Hsu, W.L.: Biomedical named entity recognition and linking datasets: survey and our recent development. Briefings in Bioinformatics **21**(6), 2219–2238 (2020)
13. Khan, M.R., Ziyadi, M., AbdelHady, M.: Mt-bioner: Multi-task learning for biomedical named entity recognition using deep bidirectional transformers. arXiv preprint arXiv:2001.08904 (2020)
14. Kim, J.D., Ohta, T., Tsuruoka, Y., Tateisi, Y., Collier, N.: Introduction to the bio-entity recognition task at jnlpba. In: Proceedings of the international joint workshop on natural language processing in biomedicine and its applications. pp. 70–75. Citeseer (2004)
15. Kitaev, N., Kaiser, Ł., Levskaya, A.: Reformer: The efficient transformer. arXiv preprint arXiv:2001.04451 (2020)
16. Kocaman, V., Talby, D.: Biomedical named entity recognition at scale. arXiv preprint arXiv:2011.06315 (2020)
17. Kolárik, C., Klinger, R., Friedrich, C.M., Hofmann-Apitius, M., Fluck, J.: Chemical names: terminological resources and corpora annotation. In: Workshop on Building and evaluating resources for biomedical text mining (6th edition of the Language Resources and Evaluation Conference) (2008)
18. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning. p. 282–289. ICML '01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
19. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics **36**(4), 1234–1240 (2020)
20. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering (2020)

21. Methley, A.M., Campbell, S., Chew-Graham, C., McNally, R., Cheraghi-Sohi, S.: Pico, picos and spider: a comparison study of specificity and sensitivity in three search tools for qualitative systematic reviews. BMC health services research **14**(1), 1–10 (2014)
22. Nye, B., Li, J.J., Patel, R., Yang, Y., Marshall, I.J., Nenkova, A., Wallace, B.C.: A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In: Proceedings of the conference. Association for Computational Linguistics. Meeting. vol. 2018, p. 197. NIH Public Access (2018)
23. Pascanu, R., Mikolov, T., Bengio, Y.: Understanding the exploding gradient problem. CoRR **abs/1211.5063** (2012), http://arxiv.org/abs/1211.5063
24. Rocktäschel, T., Weidlich, M., Leser, U.: Chemspot: a hybrid system for chemical named entity recognition. Bioinformatics **28**(12), 1633–1640 (2012)
25. Schmidt, L., Weeds, J., Higgins, J.: Data mining in clinical trial text: Transformers for classification and question answering tasks. arXiv preprint arXiv:2001.11268 (2020)
26. Stylianou, N., Razis, G., Goulis, D.G., Vlahavas, I.: Ebm+: Advancing evidence-based medicine via two level automatic identification of populations, interventions, outcomes in medical literature. Artificial Intelligence in Medicine **108**, 101949 (2020)
27. Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K.P., et al.: String v10: protein–protein interaction networks, integrated over the tree of life. Nucleic acids research **43**(D1), D447–D452 (2015)
28. Tay, Y., Dehghani, M., Bahri, D., Metzler, D.: Efficient transformers: A survey. arXiv preprint arXiv:2009.06732 (2020)
29. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30, pp. 5998–6008. Curran Associates, Inc. (2017)
30. Wang, S., Li, B., Khabsa, M., Fang, H., Ma, H.: Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768 (2020)
31. Wang, X., Zhang, Y., Ren, X., Zhang, Y., Zitnik, M., Shang, J., Langlotz, C., Han, J.: Cross-type biomedical named entity recognition with deep multi-task learning. Bioinformatics **35**(10), 1745–1752 (2019)
32. Weber, L., Münchmeyer, J., Rocktäschel, T., Habibi, M., Leser, U.: Huner: improving biomedical ner with pretraining. Bioinformatics **36**(1), 295–302 (2020)
33. Weber, L., Sänger, M., Münchmeyer, J., Habibi, M., Leser, U., Akbik, A.: HunFlair: an easy-to-use tool for state-of-the-art biomedical named entity recognition. Bioinformatics (01 2021). https://doi.org/10.1093/bioinformatics/btab042, https://doi.org/10.1093/bioinformatics/btab042, btab042
34. Yoon, W., So, C.H., Lee, J., Kang, J.: Collabonet: collaboration of deep neural networks for biomedical named entity recognition. BMC bioinformatics **20**(10), 55–65 (2019)
35. Zhang, W., Chen, Y., Liu, F., Luo, F., Tian, G., Li, X.: Predicting potential drug-drug interactions by integrating chemical, biological, phenotypic and network data. BMC bioinformatics **18**(1), 1–12 (2017)