# Named Entity Recognition in Spanish Biomedical Literature: Short Review and Bert Model

Liliya Akhtyamova
Technological University Dublin
Dublin, Ireland
akhtyamova@phystech.edu

***Abstract*—Named Entity Recognition (NER) is the first step for knowledge acquisition when we deal with an unknown corpus of texts. Having received these entities, we have an opportunity to form parameters space and to solve problems of text mining as concept normalization, speech recognition, etc. The recent advances in NER are related to the technology of contextualized word embeddings, which transforms text to the form being effective for Deep Learning. In the paper, we show how NER model detects pharmacological substances, compounds, and proteins in the dataset obtained from the Spanish Clinical Case Corpus (SPACCC). To achieve this goal, we train from scratch the BERT language representation model and fine-tune it for our problem. As it is expected, this model shows better results than the NER model trained over the standard word embeddings. We further conduct an error analysis showing the origins of models' errors and proposing strategies to further improve the model's quality.**

## I. Introduction

Started from Bag of Words (BoW), the text preprocessing has evolved to more intricate word representations such as word2vec word embeddings [1], Glove [2] and FastText [3] embeddings with the last ones being able to capture the subword information from texts. Being advanced for a range of different tasks in Natural Language Processing (NLP), methods using word embeddings gave a significant boost in model performance for biomedical NER tasks [4].

However, the most significant breakthrough in performances gave recently introduced contextualized word embeddings. Among them are Semi-supervised Sequence Learning [5], ELMo [6], ULMFiT [7], the OpenAI transformer [8], the Transformer [9], BERT [10] and Flair [11]. In our experiments, we use the BERT embeddings as they have shown to outperform other types of embeddings on a variety of sequence labeling tasks [10]. In this work, we pre-train our own BERT model using the corpus retrieved from the Scientific Electronic Library Online (SciELO) website is used to construct corpus.

Our contributions are as follows:

1) Short review focused on biomedical NER;
2) We retrieve task-specific corpora for training BERT model;
3) We train BERT language representation model from scratch and then fine-tune it for a downstream task;
4) We compare the constructed in-domain BERT model performances with the standard in-domain embeddings as well as general-domain BERT producing a new baseline;
5) Finally, we provide a careful error analysis of results.

## II. Related work

### A. Approaches to entity extraction from biomedical texts

Prior studies on biomedical entity extraction have focused primarily on the extraction of entities from English biomedical texts. In this section, we cover pharmacological entity extraction methods. The naive approach, which sometimes gives surprisingly well results is rule-based NER. Numerous research authors applied them to their problems. For example, [12] built a set of the regex to extract evidence-based dietary recommendations from scientifically validated websites' data and scientific publications. They first detected the targeted mentions in textual data, which is known as a classification task and then extracted them using the rule-based technique.

The earliest work on machine learning-based NER includes such techniques as reranking relying on both kernels [13] as well as pure feature processing [14]. Kernel-based (KB) methods for entity extraction such as Support Vector Machines (SVM) represented in numerous papers [15], [16], [16] overall became popular methods for extracting entities from texts applying as well to biomedical texts [17]. In the latter paper, the authors examined different kernel functions for the problem of biomedical NER and concluded that tree-based kernel is more capable of entity extraction.

But the most cutting-edge high-performance methods include neural network (NN)-based architectures, in particular, deep learning (DL) convolutional (CNN) and recurrent (RNN) NNs, and recently transformer deep NNs and utilizing pre-trained on larger corpora NNs for the downstream task called transfer learning. One should note that when dealing with more complex biomedical NER problems including long, discontinuous, overlapping entities hybrid approaches show the best results. For example, the authors [18] integrated KB embeddings in their tree-structured long short-term memory networks (LSTM) framework, which led to about 3% F-score gain.

In relation to contextualized word embeddings, in a paper of [19], the authors pre-trained a BERT model on a huge corpus of English biomedical texts and then fine-tuned it for three biomedical text mining tasks such as NER, relation extraction, and question answering. Their domain-specific BERT model showed 0.51% absolute improvement on biomedical NER tasks. In total, they tested their model on 15 popular biomedical corpora. Our experiments are very similar to this work in that sense that we also pre-train our own BERT embeddings and further fine-tune them for downstream problem. In another work [20], BERT embeddings were first trained

on English clinical corpora. However, the authors did not get any improvements with them on 5 clinical NER benchmark datasets over the model of [19], they got improvements on two of them when fine-tuning the model of [19] to a clinical domain tasks. They showed that in some cases, the fine-tuning could be beneficial rather than training domain-specific LM from scratch.

However, BERT is a powerful language representation there exist more light-weight and easy to train LM architectures such as Flair and ELMo. Among them, Flair is believed as more accurate. So, in a work of [21] the authors showed that domain-based contextualized word embeddings (ELMo and Flair) heavily influence the performance on downstream tasks, outperforming embeddings trained either on general-purpose data or on scientific papers when applied to user-generated content. Moreover, in their experiments they concluded that Flair embeddings perform slightly better than ELMo ones. In another paper of [22], the authors test the performances of different combinations of word embeddings in single and multi-task settings on 5 English benchmark datasets. They conclude that the combination of different types of embeddings improve NER results; moreover, as in [21] they also state that Flair embeddings on average perform better than the ELMo ones.

One can find the extensive coverage of recent advances in NLP field in the paper of Young et al. [23] with topics covered: distributed and contextualized word representations, deep learning methods and techniques and recent trends coupling deep learning models with memory modules. In another overview paper of Deng [24], the history and span of NLP progress for different applications are covered. They emphasize and set up as a fundamental framework for their whole book the next key pillars of the NLP progress: distributed and semantic representation and generalization of entities, long-span deep learning sequence language modeling, hierarchical networks for language representation, and end-to-end solutions effective to solve many NLP problems at once.

In application to biomedical NLP, a good overview of recent trends is given in [25]. They selected twelve high-quality papers, focusing on novel methods and applications of NLP over heath data touching such problems as algorithms for more precise dependency parsing of medical content, classification, and information extraction using the blend of deep learning, rule-based and knowledge aware methods, and problems of the quality of electronic medical records and textual data generated on social media space.

### B. BERT as Transformer Network

The foundation for the development of the BERT representation lies in *transformer* concept introduced in a paper of Vaswani et al.[9]. In this paper, it was shown that a deep neural network built on a pure attention mechanism achieves comparable or superior performance to the LSTM-based NN. Besides it, transformers are more computationally efficient than LSTMs as they do not require as inputs the ordered sequences of characters but instead model dependencies through *positional encoding*. It is simply relative positions of each character in a word that are concatenated with the self-attention layer and then fed to the feed-forward NN. The resulted concatenated layer is further normalized as it was shown to be effective in reducing the model training time [26].

The self-attention layer as it goes from its name for each word attends to each other word in the same sentence and learns the most relevant words for it. It is done through simple dot-product attention. However, the only difference of the conventional dot-product with that used in the transformer is applied scaled factor. Because the transformer operates on large matrices, the dot product grows large and pushes the softmax function into regions with diminishing gradient descent. To eliminate that problem, in the transformer the dot product is divided with a scale factor related to the root of the dimension.

The introduced concept of multi-head attention in transformers through different weight initialization for each head allowed to escape the local minimum for context learning and select the most relevant surrounding terms for each word. In a transformer, there are three multi-head attention networks, one in encoder which learns the word representations for the input, and the two in decoder where one learns the outputs sequence representation and the other so-called ”vanilla” network that for each output term attends and learns the most relevant input terms. Each attention layer is repeated several times forming attention blocks with non-shareable weights. In the original paper, the authors used 6 blocks but there is no optimal number for each case.

BERT is the deep learning language representation developed by Google research team [10] and stands for Bidirectional Encoder Representations for Transformers. As its name says it takes the encoded representation of the transformer as it is the main building block. Together with introduced several new hacks of learning the model, it presents a powerful way of learning language representations. Its difference from the previously introduced concept of language modeling is that it learns the context representation for word from both sides at once, while LMs such as Flair, ELMo, etc are sequential and learn left and right context of the word separately. The second difference of BERT from LMs is that it learns not a word or character embeddings but word piece and segment ones. At each training step, different word piece parts are getting masked allowing the network to learn the network to predict them. Moreover, in contrast to LM pre-trained language representation for BERT could be further *fine-tuned* for the downstream task by adding a shallow DL layer connecting to the end of the original model. Then, the model is further trained for a small number of epochs with data and labels specific to the task.

## III. Experimental Setup

### A. Dataset for BERT model training

It was shown that domain-specific contextualized word embeddings provide better results than the general domain ones for the biomedical information retrieval (IR) tasks [19], [27]. We scraped the subset of SciELO documents based on some heuristics. In particular, we parsed articles based on the next condition: the section’s area is Health Sciences and the text should be between particular strings – sections of the articles. So, the starting section should be: ’Descripcion del caso’, ’Presentacion de caso’, ’Descripcion de caso clınico’,

'Caso clınico', and the ending section should be: 'Bibliografıa', 'Referencias'. This way we retrieved 1,368,080 sentences with the number of tokens 86,851,275. This is substantially smaller than the One Billion Word corpus but provides with the in-domain corpora for training word embeddings.

We used this corpus for training BERT language representations. The vocabulary size was set to 128000 and the number of training steps 1B. BERT embeddings were trained using Tensor Processing Units (TPU) instances in Google Colab. TPU is designed to efficiently scale operations among different machines thus making calculations on tensors faster than doing it using GPU instances. However, currently, TPU does not support making predictions for downstream tasks, it is needed to switch on to CPU instances for doing it. For storing and uploading weights for training Google Cloud persistent storage is needed to be used. Moreover, every 8 hours Google Colab is shutting down its server, so it is needed to be resumed manually.

### B. Training details

The problem of biomedical NER is a sequence labeling task where the goal is to extract the correct spans of entities of 4 different types: Normalizables: mentions of concepts which can be normalized in Snomed-CT and ChEBI databases; No Normalizables: concepts from the first category which cannot be normalized to DB; Proteinas: mentions of genes and proteins; Unclear: general substance mentions. For the official evaluation, only the first 3 types of entities are used. To classify entities, we used a BIO schema. These classify entities in a document as [B]eginning, [I]nside, [O]utside.

Our BERT pre-training and fine-tuning process are similar to those used in Lee et al. [19]. We also give as inputs to our in-domain BERT model the weights of pre-trained on general-domain corpora multilingual BERT model. Then, we pre-train on Spanish biomedical literature corpora and finally fine-tune the model. We chose to use just the last hidden state of the sequence to feed into the last classification layer for fine-tuning.

In our experiments, we used the same settings as in a paper of [10] for BERT model pretraining with 12 layers, 768 hidden states, 12 attention heads and pooling operation on the first sub-word tokens of each word.

To compare our in-domain BERT model, we utilize the pre-trained multilingual cased base BERT model. We get results using these contextualized word embeddings by feeding them in the state-of-the-art Flair NER framework. For fine-tuning the BERT model, we use the default settings. The only difference of them from our model is the use of concatenated last four hidden states of the BERT model to feed into the last prediction layer.

We used a Conditional Random Fields loss [28] as it has shown to increase the accuracy for the NER tasks. The training and evaluation batch sizes were set to 32 and 8 accordingly, and the learning rate was set to 5e-5. The maximum sequence length was set to 160. Despite the common advice to fine-tune the BERT model for just 3-10 epochs, we fine-tuned it for 30 epochs as we noticed it improved the predictions. The overall architecture of BERT model is presented in Fig. 1.

## IV. Experiments

### A. Dataset for experiments

The statistics on SPACCC corpus is presented in Table I and entity class distribution in Table II. In total, we used 1000 cases. It could be seen that entity classes in the dataset are heavily unbalanced with the smallest class No_normalizables comprising roughly 1.2% of the largest Normalizables class.

Moreover, we found that around 20% of all annotated entities in the training dataset are compound entities meaning that they comprise more than 2 words split by space or dash. 80% of them are split by space.

TABLE I. Statistics on SPACCC corpus

| Size (sent) | Size (words) |
|---|---|
| 16,504 | 396988 |
| 16.5 sent/case | 396.2 words/case |

TABLE II. Entity type distribution

| Entity types | Counts |
|---|---|
| Normalizables | 4,426 |
| No_Normalizables | 55 |
| Proteinas | 2,291 |
| Unclear | 159 |

### B. Results of experiments

The results of the experiments are presented in Table III and its graphical illustration in Fig. 2. Here standard embeddings model combines FastText, (byte-pairwise encoding) BPE and character embeddings as stacking of different embeddings usually give better results. Here, however, both BPE and character embeddings are both domain indifferent, we utilized in-domain pre-trained FastText embeddings for these experiments [29]

In-domain BERT embeddings lead to better results than in-domain standard ones, especially in terms of precision. General-domain multilingual BERT embeddings with the default best settings give worse results than both in-domain standard BERT-based NER models on specialized Spanish biomedical dataset.

We also experimented with searching concepts into SNOMED-CT using Meaning Cloud tool, however, it did not work well, as many concepts for the shared task were annotated based on their synonyms.

TABLE III. Results of experiments

| | Precision | Recall | F-score |
|---|---|---|---|
| Standard embeddings | 0.87 | 0.87 | 0.87 |
| General-domain BERT | 0.86 | 0.82 | 0.84 |
| In-domain BERT | 0.90 | 0.87 | 0.89 |

## V. Error Analysis

For the error analysis, we calculated the distribution of predicted entities based on their relative to true entities' location within sentences. In section 5.1, we describe groups of errors we chose for our analysis, and in section 5.2 we present the results of this analysis.

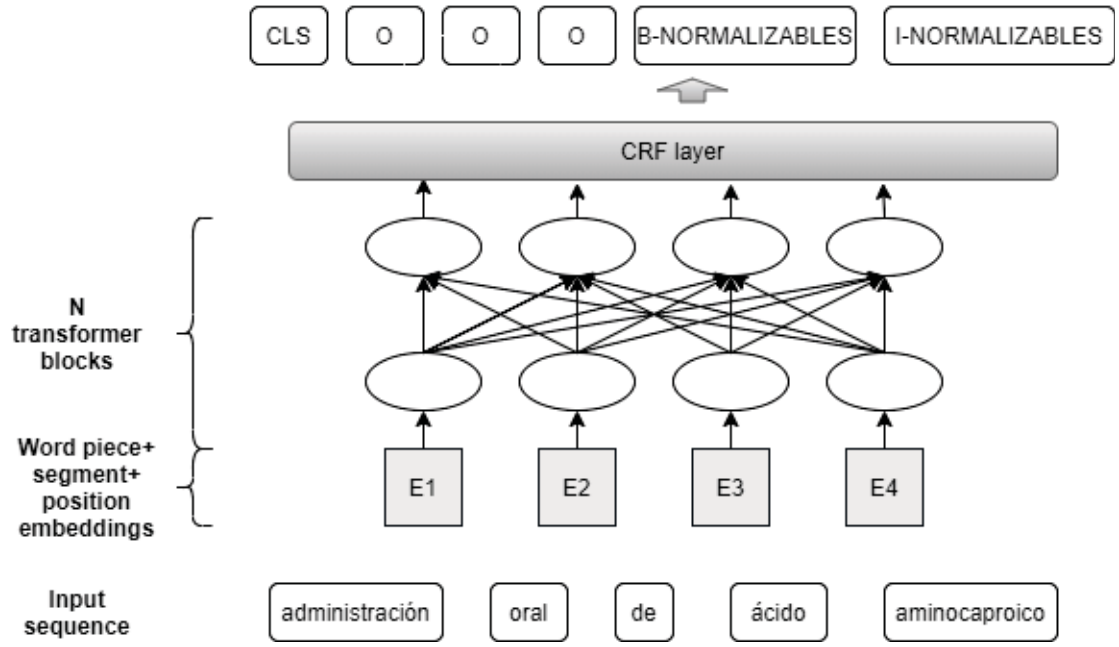


Fig. 1. BERT model architecture

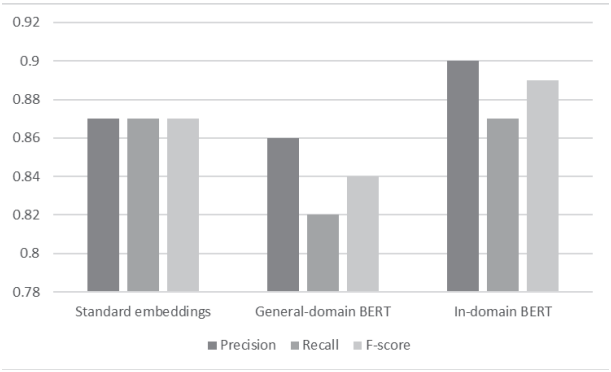


Fig. 2. Comparison of results (graphical illustration for Table III)

### A. Types of errors

We perform error analysis separately for short and long entities. Here, **short entities** are those with length 1-2 words and **long entities** have a length more or equal 3 words. Overall, types of errors could be split into the next groups with the examples of erroneous entities given for the best model, i.e. *in-domain* BERT:

- **Misrecognized entities** denote entities which models detect with the correct boundaries but gave them the wrong classes. For example, the best model misclassified "TG" entity as *Proteinas*, however, it is *Normalizables*, and labeled "urokinasa" as *Normalizables*, however its true label *Proteinas*.
- **Shorter predicted** entities are those entities which has only one matched with true entity boundary with the second boundary located within the boundaries of the true entity. Examples include predicted true label "Na+" as "Na" and "CA 15-3" as "CA 15- ". Among long entities, the best model, for example, predicted "anticuerpos inmunoglobluina M" instead of true entity "anticuerpos inmunoglobluina M (IgM) para parvovirus B19" for *Proteinas* entity.
- **Longer predicted** entities denote longer predicted entities with matched one boundary. For example, in some cases the true entity had to be "Na", but models wrongly predicted "Na+", or the model predicted "biotina peroxidasa" instead of "peroxidasa".
- **Intersected boundaries** predicted entities having either left end or right end within the true entity but not with any equal borders. An example is the true entity "EBV Ac antianticípside IgM" with predicted entity "Ac antianticípside".
- **Different boundaries** denote newly detected cases where the predicted entities are not found to be among gold standard entities for a current sentence at all. Examples are "isoenzimas de FA" predicted by the best model as a new *Proteinas* entity, and "levobupivacana" as a new *Normalizables* entity. These entities are not found to be gold standard entities in the sentences where they were predicted as such.
- **Not detected** entities are those not detected by models at all. For example, the best model did not predict "fingolimod", "aprepitant", "PRP" *Normalizables* entities, did not predict "factores de crecimiento de PRP", "VII", "pS" *Proteinas* entities and did not predict any *No_normalizables* entity.

For the classes of errors above, in data mining the first

TABLE IV. ERRORS DISTRIBUTION FOR SHORT ENTITIES

| | Misrecognized entities | Shorter predicted | Longer predicted | Intersected boundaries | Different boundaries | Not detected |
|---|---|---|---|---|---|---|
| Standard embeddings | 30 | 22 | 58 | 3 | 103 | 72 |
| General-domain BERT | 33 | 50 | 32 | 0 | 33 | 125 |
| In-domain BERT | 27 | 48 | 31 | 1 | 30 | 110 |

TABLE V. ERRORS DISTRIBUTION FOR LONG ENTITIES

| | Misrecognized entities | Shorter predicted | Longer predicted | Intersected boundaries | Different boundaries | Not detected |
|---|---|---|---|---|---|---|
| Standard embeddings | 2 | 31 | 1 | 0 | 6 | 1 |
| General-domain BERT | 0 | 36 | 0 | 0 | 4 | 8 |
| In-domain BERT | 0 | 32 | 0 | 0 | 3 | 6 |

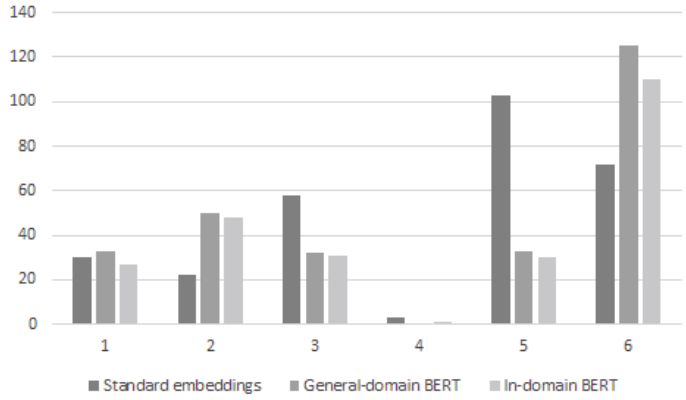


Fig. 3. Error analysis for short predicted entities (graphical illustration for Table IV)

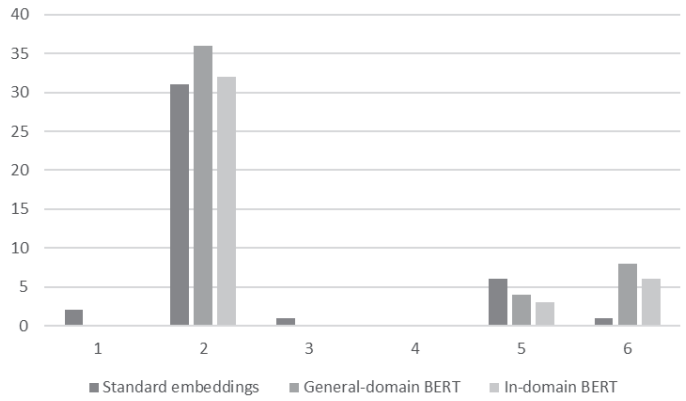


Fig. 4. Error analysis for long predicted entities (graphical illustration for Table V)

5 types of errors are called *false positives* (FP), and the last group of errors is called *false negatives* (FN).

### B. Results of analysis

The results of this analysis are presented in Table IV for short predicted entities, and in Table V for long predicted entities, and in corresponding Pictures 3 and 4 accordingly. In these pictures, numbers on x-axis denote:

1: Misrecognized entities

2: Shorter predicted

3: Longer Predicted

4: Intersected boundaries

5: Different boundaries

6: Not detected

The biggest benefit of contextualized models in application to our dataset is linked to making less mistakes in recognizing FP with no intersected boundaries. However, it is still observable that it makes more mistakes in not recognizing true entities (FN) while the standard embedding-based model performs much better here.

The next source of errors for the in-domain BERT-based model is shorter detected entities followed by misrecognized, longer predicted and different boundaries entities' errors in roughly the same proportions.

## VI. Conclusions

In the paper, we present a short review related to the application of different techniques to the extraction of biomedical entities. It clearly shows the advantage of contextualized pre-trained language representations for solving numerous NLP problems. However, their benefit for low-source languages such as Spanish is not fully studied yet, especially for the biomedical domain.

In our experiments, we train the domain-specific Spanish BERT-based contextualized word embeddings. Then, the BERT model is fine-tuned for the downstream task. We show that domain-specific contextualized word embeddings outperform both the domain-specific standard and general-domain BERT-based embeddings, however, trained on a smaller corpus.

To the best of our knowledge, our work is the first to release BERT model trained on Spanish biomedical texts.

Additionally, we perform an error analysis investigating the sources of models' mistakes. We discover that our best in-domain BERT model makes the majority of errors by not recognizing gold standard entities. In contrast to it, the model with the standard embeddings makes the majority of mistakes in recognizing the false positives new entities. We suspect that the efficient combination of both types of embeddings would reduce these sources of errors and improve the model accuracy. This part is left for future experiments.

It would be also interesting to conduct more extensive experiments with other types of LMs, ie in addition to pre-training BERT from scratch on the Spanish biomedical texts, pre-train from scratch, for example, ELMo or Flair embeddings which in many applications has shown its competitive performance.

Moreover, the utilization of more sophisticated architectures for training NER model with the dependency graph attention mechanism integrated into a model could further benefit the model performance.

## Acknowledgment

The author thanks Dr. Mikhail Alexandrov (RANEPA, Russia) and Dr. John Cardiff (TU Dublin, Ireland) for the helpful advices, which promoted more extended problem settings. The author also thanks Paloma Martinez (University Carlos III, Spain) and Karin Verspoor (University of Melbourne, Australia) for discussions concerning experiments and interpretations of their results.

The author wants to express his gratitude to the anonymous reviewers of this work. whose critical (and very critical) comments contributed to an improvement in the quality of the article.

## References

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[2] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.

[3] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, vol. 5, no. 2307-387X, pp. 135–146, 7 2017.

[4] M. Habibi, L. Weber, M. Neves, D. L. Wiegandt, and U. Leser, "Deep learning with word embeddings improves biomedical named entity recognition," *Bioinformatics*, vol. 33, no. 14, pp. i37–i48, 7 2017.

[5] A. M. Dai and Q. V. Le, "Semi-supervised Sequence Learning," 11 2015.

[6] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2 2018.

[7] J. Howard and S. Ruder, "Universal Language Model Fine-tuning for Text Classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 328–339.

[8] A. Radford, "Improving Language Understanding by Generative Pre-Training," in *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf.*, 2018.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *In Advances in neural information processing systems*, pp. 5998–6008, 6 2017.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *arXiv preprint arXiv:1810.04805*, 10 2018.

[11] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual String Embeddings for Sequence Labeling," in *COLING*, 2018.

[12] T. Eftimov, B. Korouš Seljak, and P. Korošec, "A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations," *PLoS ONE*, vol. 12, no. 6, 2017.

[13] T.-V. T. Nguyen, A. Moschitti, and G. Riccardi, "Kernel-based reranking for named-entity extraction," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics (ACL), 2010, pp. 901–909.

[14] M. Collins, "Ranking algorithms for named-entity extraction," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics (ACL), 2001, pp. 489–496.

[15] J. Björne and T. Salakoski, "Generalizing Biomedical Event Extraction," in *Proceedings of BioNLP Shared Task 2011 Workshop*, 2011, pp. 183–191.

[16] H. Isozaki and H. Kazawa, "Efficient support vector classifiers for named entity recognition." Association for Computational Linguistics (ACL), 2002, pp. 1–7.

[17] R. Patra and S. K. Saha, "A kernel-based approach for biomedical named entity recognition," *The Scientific World Journal*, vol. 2013, 2013.

[18] D. Li, L. Huang, H. Ji, and J. Han, "Biomedical Event Extraction Based on Knowledge-driven Tree-LSTM," in *Proceedings of NAACL-HLT 2019*. Association for Computational Linguistics, 2019, pp. 1421–1430.

[19] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, no. btz682, 2019.

[20] E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, and M. B. A. McDermott, "Publicly Available Clinical BERT Embeddings," in *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, 4 2019, p. 72–78.

[21] M. Basaldella and N. Collier, "BioReddit: Word Embeddings for User-Generated Biomedical NLP," in *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*. Association for Computational Linguistics (ACL), 11 2019, pp. 34–38.

[22] L. Akhtyamova and J. Cardiff, "LM-based Word Embeddings Improve Biomedical Named Entity Recognition: a Detailed Analysis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Bioinformatics) [to be published in June 2020]*. Springer Verlag.

[23] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in

deep learning based natural language processing [Review Article]," pp. 55–75, 8 2018.

[24] L. Deng and L. Yang, *Deep Learning in Natural Language Processing*. Springer Singapore, 2018.

[25] V. G. V. Vydiswaran, Y. Zhang, Y. Wang, and H. Xu, "Special issue of BMC medical informatics and decision making on health natural language processing," *BMC Medical Informatics and Decision Making*, vol. 19, no. S3, p. 76, 4 2019.

[26] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," *ArXiv*, vol. abs/1607.0, 7 2016.

[27] G. Sheikhshab and A. Sarkar, "In-domain Context-aware Token Embeddings Improve Biomedical Named Entity Recognition," in *In Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, 2018, pp. 160–164.

[28] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289.

[29] F. Soares, M. Villegas, A. Gonzalez-Agirre, M. Krallinger, and J. Armengol-Estapé, "Medical Word Embeddings for Spanish: Development and Evaluation," in *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, 2019, pp. 124–133.