The Power of BioBERT

Text Mining — Course paper

Ruben Ahrens
s3677532@umail.leidenuniv.nl
LIACS, Leiden University
Leiden, Netherlands

Lucas de Wolff
s3672980@umail.leidenuniv.nl
LIACS, Leiden University
Leiden, Netherlands

ABSTRACT

This paper investigates the effectiveness of BERT and BioBERT in biomedical text mining, with a specific emphasis on Named Entity Recognition (NER). Through fine-tuning on the CADEC dataset, consisting of medical reviews, BioBERT, specialized in biomedical texts, demonstrates superior NER performance compared to BERT. The study underscores the significance of pre-training on domain-specific data for improved NER outcomes.

KEYWORDS

Text Mining, Named Entity Recognition, Biomedical Text Analysis, BERT, BioBERT, Hyperparameter Optimization

ACM Reference Format:
Ruben Ahrens and Lucas de Wolff. 2024. The Power of BioBERT: Text Mining — Course paper. In *Proceedings of Text Mining Course 2023/2024 (Textming '24).* ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

Named entity recognition (NER) is a fundamental task in natural language processing (NLP) that plays a crucial role in various downstream applications, including information extraction, machine translation, and question-answering. The core objective of NER is to identify and classify named entities (NEs) within a given text, where NEs typically represent proper nouns such as people, places, organizations, and events.

Conventional NER approaches often employ hand-crafted features and rule-based systems, which can be computationally expensive, may not effectively scale to handle large datasets, and may not generalize well to unseen data. In recent years, deep learning models, specifically transformer-based models like BERT (Bidirectional Encoder Representations from Transformers), have emerged as powerful tools for NER, achieving state-of-the-art performance on various NER tasks. BERT models are pre-trained on large amounts of unlabeled text data and can learn representations of words and phrases that capture complex linguistic relationships.

Since its publication, BERT has attracted significant attention and is now applied across various research domains, including the biomedical field. However, complex and domain-specific texts can cause a shift in the corpora compared to BERT's original pre-training corpora. This may cause BERT to underperform. Therefore, a common strategy is to pre-train BERT a second time on domain-specific text data. BioBERT [12] leverages this concept by pre-training BERT specifically on biomedical domain texts, resulting in a substantial improvement in model performance when applied to biomedical text data for a variety of text mining tasks.

This paper conducts a comparative analysis between BERT and BioBERT in their application to biomedical text data for the task of NER. The evaluation is performed using the CADEC dataset [11], a corpus of medical review text. Each review within this dataset has been carefully annotated by domain experts, highlighting entities such as diseases, drugs, symptoms, findings, and adverse drug reactions (ADRs). To maximize the potential of the models, we utilize Optuna [1], an open-source hyperparameter optimization (HPO) library. We optimize five key hyperparameters for both BERT models: learning rate, Adam beta 1, Adam beta 2, weight decay, and Adam epsilon.

We believe that this study will contribute to a better understanding of the impact of pre-training BERT on domain-specific data on NER performance and provide valuable insights for developing more accurate and efficient NER systems for medical text analysis.

The paper will follow the outlined structure: Initially, we establish the groundwork for our research and articulate the motivation for this study in the related work section (Chapter 2). Subsequently, we explore the CADEC dataset, providing insights into its statistical properties and addressing challenges inherent to its characteristics (Chapter 3). Following this, we elaborate on the methodologies employed in our study, covering aspects such as data preparation, the HPO phase, the fine-tuning process of BERT/BioBERT, and the evaluation metrics utilized (Chapter 4). Following this, we present the outcomes of our experiments (Chapter 5), concluding with a comprehensive discussion in the final chapter (Chapter 6).

2 RELATED WORK

In this section, we explore key contributions in the field, from traditional methods to recent advancements in deep learning models, establishing the foundation for our study.

NER [13] is a pivotal task in NLP that involves the identification and categorization of named entities in text into predefined classes. Conventional methods for NER frequently depended on manually crafted features and rule-based systems. Despite their historical use, these methods present drawbacks such as computational intensity, limited scalability with large datasets, and challenges in generalizing to novel data.

The emergence of deep learning has revolutionized the field of NLP, including NER. One of the early influential models was the

Textming '24, Master CS, Leiden, the Netherlands

Long Short-Term Memory (LSTM) [9], which is an extension of the Recurrent Neural Network (RNN). LSTMs are well-suited for sequential data like text, making them effective for tasks such as NER. However, the real potential of LSTM and its bidirectional version (BiLSTM) became evident with the introduction of Bidirectional LSTM-CRF Models for Sequence Tagging [10].

The next big breakthrough was with the introduction of the transformer model [15]. Transformer-based models, such as BERT [5], have proven to be particularly effective, achieving state-of-the-art performance on various NER tasks. Their attention mechanism enables these models to effectively account for long-range dependencies and comprehend complex semantic relations.

However, the performance of BERT is less consistent when applied to corpora that deviate significantly from its original training data. In response to this challenge, the DMISLAB at Korea University introduced BioBERT [12]. This model is specifically designed to enhance BERT's performance when dealing with biomedical text data.

BiomedBERT, developed for a similar purpose, represents an innovative approach [7]. This model diverged by undertaking a comprehensive retraining of BERT from scratch, utilizing abstracts sourced from PubMed and full-text articles from PubMedCentral. Impressively, BiomedBERT nearly surpassed state-of-the-art models on the Biomedical Language Understanding and Reasoning Benchmark (BLURB) [8]. As of the latest benchmark standings, BioLinkBERT [18] currently holds the top position on BLURB. Notably, unlike conventional models such as BERT that focus on modeling a single document, LinkBERT demonstrates the capability to capture dependencies and knowledge that extend across multiple documents.

Our research makes a meaningful contribution to the existing literature by empirically assessing the performance enhancement of BioBERT in comparison to BERT on a biomedical dataset like CADEC. We anticipate that our study will yield valuable and compelling insights.

3 DATA

The CSIRO Adverse Drug Event Corpus (CADEC) dataset is an annotated corpus of medical forum posts where patients report Adverse Drug Events (ADEs). To be precise, the reviews were extracted from askapatient.com. The dataset's annotations include three label categories: SnowmedCT, MedDra, and original named entity recognition labels. For this study, the original NER-labeled data was used. This version contained a variety of labels: ADR, Symptom, Drug, Disease, and Finding:

- *ADR* stands for adverse drug reaction, a reaction to the body of the patient after taking the drug, that is unwanted.
- *Symptom* signifies the underlying health issue or condition that prompts the patient to use the drug.
- *Drug* is assigned any time a word refers to a drug, apart from some specific drug classes and medical devices.
- *Disease* is assigned whenever the name of a disease is mentioned and specifies the reason for taking the drug.
- *Finding* is described as any adverse side effect, disease, or symptom that was not directly experienced by the patient itself. It also includes any other clinical concept that could potentially fall into these categories, with the annotator uncertain about its specific classification.

In table 1, the label distribution given for the training, validation, and test set is shown. Note that the classes aren't equally represented in the dataset. As expected of a patient ADEs forum, ADR is the most represented class, followed by the Drug category. In the last column, the number of unique entities of each class is shown. The ratio of total entities to unique entities also vastly differs between different labels.

Entity	All	Unique
ADR	6318	2713
Disease	283	162
Drug	1800	321
Symptom	275	130
Finding	435	265
All	9111	3591

Table 1: Entity distribution of the CADEC dataset and the number of unique entities as described in [11].

The data itself is separated on each drug and has its corresponding reviews. Some drugs are more well-known or prescribed than others, and thus, appear more often in the dataset. Table 2, shows the number of reviews per drug.

Name	Frequency
Voltaren	46
Flector	1
Arthrotec	145
Diclofenac-Potassium	3
Lipitor	1000
Diclofenac-Sodium	7
Cataflam	10
Pennsaid	4
Cambia	4
Zipsor	5
Voltaren-XR	22
Solaraze	3
Total	1250

Table 2: Number of documents per drug for the CADEC dataset and total number of documents.

4 METHOD

This section details the methodological approach employed in this research. First, the data underwent some necessary preprocessing. Then, both BERT's and BioBERT's model parameters were optimized using the Optuna library. With the newly optimized parameters, the models were adapted to the CADEC dataset for the task of NER.

4.1 Data Preparation

4.1.1 Tokenization. As the first data preprocessing step, the labels had to be extracted from the annotation files. The annotation files

only contained the character positions of entities of interest linked with their corresponding labels. The challenge was to map the original review text to its named entities, including the "O" label for no entity. This was achieved by inserting a unique starting and ending tag around each of the named entities based on the positions provided in the annotation. For example, the text "I began having cramps" with the annotation "ADR 15-21" is first transformed into "I began having <ADR> cramps </ADR>". Then, the starting and ending tags were added to the tokenizer vocabulary, to prevent the tokenizer from splitting the tags into subwords. After this, the text was tokenized using the BERT base uncased tokenizer from the transformers library. The example text is then tokenized as: ["i", "began", "having", "<ADR>", "cramps", "</ADR>"]. Then finally, this is automatically transformed into the series ["i", "began", "having", "cramps"] and ["O", "O", "O", "B-ADR"], following the IOB2-tagging scheme [14]. An edge case was encountered during this process, where some words would have multiple labels assigned. In this case, we would simply pick the first assigned label and ignore the other labels to avoid a multi-label classification task.

4.1.2 Handling special cases. In figure 1, the distribution of the token length is plotted. Most documents contain around 100 tokens. Out of the 1250 reviews, 4 reviews had a token length of more than 512 tokens. Since BERT is not able to handle input sequences with more than 512 tokens, we chose to simply eliminate these reviews.

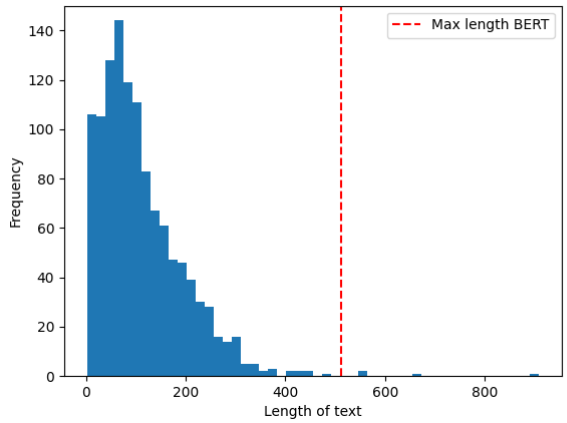

Figure 1: Distribution of document lengths on token-level of CADEC data.

Additionally, there were reviews without any annotations, 64 to be exact. We chose to exclude them as they provide less valuable information for the models compared to other files. However, it's essential to note that even non-annotated reviews can contain valuable insights, and the decision to discard them was based on specific considerations. After discarding these reviews, the final count of the remaining reviews was 1182.

4.1.3 Splitting the data. Next, the data was split into a train (80%), validation (10%), and test set (10%). To address potential bias towards specific medicine types in the dataset, a stratified split was implemented based on individual medicine types. This approach made sure each data subset roughly had the same distribution of medicines. Within each split, the individual reviews for each medicine were further shuffled to preserve randomness within each subset. This two-step process ensured both fair representation of different medicines and maintained sufficient randomness within each split.

4.2 BERT & BioBERT

For our BERT model, a pre-trained version was acquired through Hugging Face Transformers [17], called 'bert-base-cased' [4]. The datasets used to pre-train this model consist of Wikipedia [6] and BookCorpus [19]. It was pre-trained on the task of masked language modeling, where the model has to predict a hidden word within a sentence or text, and on next sentence prediction (NSP).

The BioBERT model has copied the 'bert-base-cased' model parameters but has been pre-trained a second time on PubMed [2] abstracts and PMC Full-text articles [3]. This is expected to result in a better performance on the CADEC dataset since it is pre-trained on a similar corpora. The model *biobert-v1.1* is available on the Huggingface transformers library.

4.3 Hyperparameter optimization

Hyperparameter optimization can significantly improve a model's performance. In this study, the Optuna optimization library was used to maximize the metrics of BERT and BioBERT. Both were separately optimized for 10 trials, each max 2 epochs long. The optimized hyperparameters include learning rate, Adam beta 1, Adam beta 2, weight decay, and Adam epsilon. Optuna's default TPE [16] sampler was used. Through experimentation, the precise search ranges and sample distributions were established for each hyperparameter.

4.4 Fine-tuning

Fine-tuning is a necessary step when leveraging pre-trained transformer models for a downstream task like NER. As both BERT and BioBERT were not pre-trained on the task of NER, they were fine-tuned on the CADEC dataset for this task. The hyperparameters found using Optuna were used, as well as a batch size of 4 and a gradient accumulation interval of 2 to avoid CUDA OOM errors. The models were trained for 5 epochs, after which they were evaluated using the metrics described in the following subsection. While the original BERT paper suggested a range of 2 to 4 epochs for optimal performance across various tasks, our experimentation revealed that 5 epochs achieved the best results for this dataset.

4.5 Evaluation Metrics

To assess the performance of both models, the following metrics were computed using the evaluate[1] library: precision, recall, F1-score, and accuracy. During HPO, the overall precision, recall, F1-score, and accuracy were maximized. During fine-tuning, the same metrics were used to compute the loss. In the final reporting of results, metrics were calculated individually for each entity type

[1] https://huggingface.co/docs/evaluate/index

(ADR, disease, drug, symptom, finding) and for the distinct B and I tags.

4.6 Software and Libraries

The research leveraged the following software and libraries:

- Hugging Face Transformers (backend: PyTorch) for model training and evaluation [17]
- Optuna for hyperparameter optimization [1]

5 RESULTS

In this section, the results of the study will be presented. Firstly, the results of the hyperparameter optimization and its corresponding optimal values are presented in table 3 and 4.

Hyperparameter	Range	Optimal value
Learning rate	1e-6 ↔ 1e-3	7.940e-5
Adam beta1	0.9 ↔ 0.999	0.929
Adam beta2	0.9 ↔ 0.999	0.929
Weight decay	0.0 ↔ 0.1	0.036
Adam epsilon	1e-9 ↔ 1e-7	7.771e-8

Table 3: Seach space and best HP values for BERT

Found by Optuna's TPE algorithm after 10 trials, each run for max 2 epochs.

Hyperparameter	Range	Optimal value
Learning rate	1e-6 ↔ 1e-3	1.481e-4
Adam beta1	0.9 ↔ 0.999	0.950
Adam beta2	0.9 ↔ 0.999	0.950
Weight decay	0.0 ↔ 0.1	0.081
Adam epsilon	1e-9 ↔ 1e-7	1.070e-8

Table 4: Seach space and best HP values for BioBERT

Found by Optuna's TPE algorithm after 10 trials, each run for max 2 epochs.

Table 5 presents the results for BERT, while the outcomes for the individual B and I tags are detailed in table 6. Likewise, the results for BioBERT are presented in table 7, with the individual B and I tags shown in table 8.

In table 9, the micro-averages of BERT and BioBERT are presented side by side for clear comparison.

	Average precision	Average recall	Average F1
BERT	0.64	0.63	0.64
BioBERT	**0.70**	**0.71**	**0.70**

Table 9: Comparison of BERT and BioBERT Micro averages.

6 DISCUSSION & CONCLUSION

In Table 9, it is evident that BioBERT consistently outperforms BERT across all metrics, aligning with expectations due to its more specialized pre-training for the task. However, an examination of individual entities in Tables 5 and 7 reveals that BERT achieves comparable results on the 'Finding' entity. This might be attributed to the inherent ambiguities associated with this entity type, making it challenging for both the model and annotators. In such cases, BioBERT does not have an edge over BERT and performs similarly.

For both models, entities with lower performance tend to have smaller support values. A more evenly distributed dataset might have led to improved results. However, it could also be attributed to the inherent nature of certain entities; some entities might inherently be more challenging to predict or contain more noise.

Table 6 and table 8, show that BERT can perform fairly well on the individual B/I tags for some entities. However, it is apparent that BERT has more trouble with classifying the I tags correctly compared to the B tags. One theory could be that the B tags are often more obvious and contain more information about the entity compared to the I tags.

Although we showed the promise of using task-specific transformer models, there are several improvements possible for future work. The corpus contained several documents longer than the maximal recommended length of 512 for BERT transformer models. Given more time we could have prevented leaving out these documents and found a more sophisticated way to handle these documents.

With TPE, it is optimal to have a large number of trials to better enable the model to find optimal hyperparameters. The number of trails would ideally be increased for a better-performing hyperparameter configuration, such as 100. Also, each trial was trained for only 2 epochs. While 2 epochs should be enough according to the authors of the original BERT, a larger number of epochs could increase the likelihood of finding good hyperparameter values.

The stochastic nature of transformer models can significantly affect the overall performance of the models, emphasizing the importance of validation techniques. One way to combat this would be to perform k-fold cross-validation.

Since BioBERT's proposal, new ideas have been applied to improve the application of transformer models to NLP tasks. One example of this is BioLinkBERT, which is, in contrast to the original BERT, able to model knowledge that spans over multiple documents. We recommend BioLinkBERT as a superior alternative to BioBERT and see promising potential in its application to the CADEC dataset.

This study served as an empirical comparison between BERT and BioBERT, fine-tuned for NER on CADEC, a medical dataset. We found BioBERT to outperform BERT on this task, making use of its pre-training on medical data. This highlights the potential benefits of pretraining transformer models like BERT on domain-specific data when challenged with domain-specific tasks.

Table 5: BERT results for grouped entities (test set).

	Precision	Recall	F1	Support
ADR	0.63	0.64	0.64	790
Disease	0.23	0.25	0.24	36
Drug	0.88	0.91	0.89	208
Finding	0.30	0.21	0.25	67
Symptom	0.25	0.11	0.16	44
Micro	0.64	0.63	0.64	1145
Macro	0.46	0.43	0.43	1145

Table 6: BERT results for B/I tags (test set).

	Precision	Recall	F1	Support
B	0.76	0.78	0.77	790
I	0.81	0.79	0.80	2595
B	0.32	0.36	0.34	36
I	0.54	0.44	0.48	234
B	0.93	0.97	0.95	208
I	0.93	0.97	0.95	1065
B	0.45	0.31	0.37	67
I	0.38	0.29	0.33	326
B	0.35	0.16	0.22	44
I	0.19	0.10	0.13	115

Table 7: BioBERT results for grouped entities (test set).

	Precision	Recall	F1	Support
ADR	0.66	0.70	0.68	790
Disease	0.69	0.72	0.70	141
Drug	0.92	0.94	0.93	208
Finding	0.29	0.22	0.25	67
Symptom	0.86	0.57	0.69	84
Micro	0.70	0.71	0.70	1290
Macro	0.68	0.63	0.65	1290

Table 8: BioBERT results for B/I tags (test set).

	Precision	Recall	F1	Support
B	0.76	0.80	0.78	790
I	0.81	0.80	0.81	2524
B	0.69	0.72	0.71	141
I	0.35	0.21	0.26	80
B	0.95	0.98	0.96	208
I	0.91	0.89	0.90	1166
B	0.37	0.28	0.32	67
I	0.35	0.38	0.36	362
B	0.88	0.58	0.70	84
I	0.90	0.16	0.26	58

REFERENCES

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. arXiv:1907.10902 [cs.LG]

[2] Kathi Canese and Sarah Weis. 2013. PubMed: the bibliographic database. *The NCBI handbook* 2, 1 (2013).

[3] Europe PMC Consortium. 2015. Europe PMC: a full-text literature database for the life sciences and platform for innovation. *Nucleic acids research* 43, D1 (2015), D1042–D1048.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL]

[6] Wikimedia Foundation. [n. d.]. *Wikimedia Downloads.* https://dumps.wikimedia.org

[7] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. arXiv:arXiv:2007.15779

[8] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare* 3, 1 (Oct. 2021), 1–23. https://doi.org/10.1145/3458754

[9] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-term Memory. *Neural computation* 9 (12 1997), 1735–80. https://doi.org/10.1162/neco.1997.9.8.1735

[10] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. arXiv:1508.01991 [cs.CL]

[11] Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015. Cadec: A corpus of adverse drug event annotations. *Journal of biomedical informatics* 55 (2015), 73–81.

[12] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36, 4 (2020), 1234–1240.

[13] Arya Roy. 2021. Recent Trends in Named Entity Recognition (NER). arXiv:2101.11420 [cs.CL]

[14] Chul Sung, Vaibhava Goel, Etienne Marcheret, Steven Rennie, and David Nahamoo. 2021. CNNBiF: CNN-based Bigram Features for Named Entity Recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, Punta Cana, Dominican Republic, 1016–1021. https://doi.org/10.18653/v1/2021.findings-emnlp.87

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* abs/1706.03762 (2017). arXiv:1706.03762 http://arxiv.org/abs/1706.03762

[16] Shuhei Watanabe. 2023. Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance. arXiv:2304.11127 [cs.LG]

[17] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art

Natural Language Processing. arXiv:1910.03771 [cs.CL]
[18] Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining Language Models with Document Links. arXiv:2203.15827 [cs.CL]
[19] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In *The IEEE International Conference on Computer Vision (ICCV)*.