Evaluating Few-Shot Prompting Approach Using GPT4 in Comparison to BERT-Variant Language Models in Biomedical Named Entity Recognition

Kranthi Kumar Konduru
School of Computer Science and Mathematics
Liverpool John Moores University
Liverpool, United Kingdom
k.k.konduru@2023.ljmu.ac.uk

Friska Natalia
Information Systems Department
Universitas Multimedia Nusantara
Tangerang, Indonesia
friska.natalia@umn.ac.id

Sud Sudirman
School of Computer Science and Mathematics
Liverpool John Moores University
Liverpool, United Kingdom
s.sudirman@ljmu.ac.uk

Dhiya Al-Jumeily
School of Computer Science and Mathematics
Liverpool John Moores University
Liverpool, United Kingdom
d.aljumeily@ljmu.ac.uk

***Abstract*— The wealth of information associated with the exponential increase in digital text, particularly within the biomedical field, has the potential to advance medical research, improve patient care, and enhance public health outcomes. However, the sheer volume and complexity of this data necessitate advanced computational tools for effective processing and analysis. We investigated the use of various pre-trained transformer-based language models, particularly BERT, PubMedBERT, SciBERT, ClinicalBERT, DistilBERT, and the application of prompt engineering with GPT-4, within the context of biomedical Named Entity Recognition. Our approach incorporates a comprehensive performance evaluation analysis utilizing standard NLP evaluation metrics and computational resource usage metrics such as training time, memory usage, and inference time. Through this multifaceted approach, we sought to find out how the few-shot prompting approach using GPT4 performs in comparison to the BERT-variant language models while at the same time identifying models that not only excel in performance efficiency but also demonstrate computational affordability. Our experimental results show that even the most basic transformer-based language model outperforms the few-shot prompting approach of GPT-4, despite the popularity of the LLM in the more general Natural Language Processing tasks.**

Keywords*— *Name Entity Recognition, Natural Language Processing, Transformer-based Language Model, Large Language Model, Biomedical Texts.

I. Introduction

The exponential increase in digital text, particularly within the biomedical field, presents both an opportunity and a challenge. On one hand, the wealth of information has the potential to advance medical research, improve patient care, and enhance public health outcomes. On the other hand, the sheer volume and complexity of this data necessitate advanced computational tools for effective processing and analysis. This is where Natural Language Processing (NLP), and more specifically, Named Entity Recognition (NER), becomes essential. NER is a subtask of NLP that involves identifying and categorizing key pieces of information in text, such as names of people, organizations, or in the case of biomedical NER, drug names, diseases, and treatment procedures. The importance of NER in biomedical research cannot be overstated. It enables the automated extraction of critical medical information from unstructured text sources, such as clinical notes, research articles, and electronic health records.

The advent of deep learning has led to substantial advancements in pre-trained language model capabilities. The introduction of pre-trained transformer-based large language models such (LLMs) as the Bidirectional Encoder Representations from Transformers (BERT) has set new benchmarks in the field. These models, which are trained on vast amounts of general or domain-specific text, can be fine-tuned in a specific domain for domain-specific NER tasks, yielding unprecedented accuracy and efficiency [1]. Several recent studies [2], [3] have also highlighted various efforts to adapt generative LLMs for these tasks, shedding light on both the potential and challenges of these endeavors. These models use natural language texts called prompts as inputs. Prompts can be in the form of questions or statements and can be as simple as a phrase or as complex as multiple sentences or paragraphs. The quality of the outputs of generative LLMs, i.e., how relevant and useful the outputs are, depends on the prompts and they can be improved by iteratively refining the prompts. When no refinement is used, the approach is called zero-shot prompting as opposed to few-shot prompting which uses a small number iteration. The process of designing the prompts to guide the LLMs to generate more accurate and relevant outputs is called prompt engineering.

To the best of our knowledge, there is very little work has been done that compares the performance of the two different language models in biomedical-domain-specific NER. In this paper, we present our study that assesses the performance of using the few-shot prompting of an LLM and model adaptation of transformer models to identify and categorize key pieces of information in biomedical texts.

II. Literature Review

Our review of the literature reveals substantial progress in the development of pre-trained bidirectional transformer-based language models with models like BioBERT [4] and PubMedBERT [5] achieving notable success in extracting biomedical entities. While these advancements demonstrate

the potential of pre-trained language models, they also highlight the gap in research concerning the computational efficiency of such systems. Case in point, studies by [6] introduced BERT, showcasing its capabilities but also its substantial resource requirements. The application of these models in clinical contexts is further explored by Huang et al. when developing ClinicalBERT while underlining the need for more computational resources [7]. The existing body of research, while advancing the accuracy and scope of biomedical NER, has not sufficiently addressed the challenges of deploying these technologies in resource-constrained settings. Moreover, the literature shows a lack of comprehensive strategies to balance the computational demands with the performance needs of biomedical NER models, particularly in diverse and rapidly evolving medical fields. Han et al. [8] identified four key directions of progress driven by advances in computational power and data availability when reviewing recent breakthroughs in pre-trained transformer-based language models. They are designing effective architectures, utilizing rich contexts, improving computational efficiency, and conducting interpretation and theoretical analysis.

Our literature review also found various efforts to adapt LLMs for domain-specific applications, shedding light on both the potential and challenges of these endeavors. In their paper, Ling et al. [2] provide a comprehensive taxonomy for categorizing domain-specialization techniques, crucial for understanding how LLMs can be tailored to specific fields. However, they also highlight critical limitations such as the inaccessibility of LLM architecture and the lack of standardized evaluation methods, which impede the widespread application of these models, particularly for those without extensive AI expertise. One such model, ClinicalGPT, is an LLM that has been specifically tailored for clinical applications in the healthcare domain [9]. However, the authors noted that, despite the general effectiveness of large language models in NLP, their application in medical settings has been challenging due to issues like factual inaccuracies and lack of domain-specific understanding. There is also TrialGPT [10] that adapts LLMs to the specific domain of clinical trials, demonstrating how general-purpose LLMs can be fine-tuned to perform specialized tasks such as predicting eligibility criteria based on patient notes.

There have been numerous efforts to address some of these challenges associated with using LLMs for domain-specific applications. For example, to address the challenge of applying data-centric approaches, such as natural language prompting, to biomedical language modeling due to the underrepresentation of labeled biomedical datasets in existing data collections, several researchers introduced BigBIO, a comprehensive community library that contains over 126 biomedical NLP datasets across 12 task categories and more than ten languages [11]. Other researchers explored tools like LMTuner [12] and QA-LoRA [13] which aim to simplify and economize the adaptation of LLMs. These studies contribute to the field by offering solutions to lower the barriers to LLM training and fine-tuning. Nevertheless, they stop short of providing comprehensive metrics on computational resources, an area critical to the theme of resource efficiency. There is also an effort to explore the Parameter-Efficient Fine-Tuning (PEFT) methods in [14] to provide insights into hardware constraints management. However, the study found that PEFT methods generally underperform compared to full model tuning in resource-limited settings further underscoring the complexity of achieving efficiency without sacrificing performance.

The work of Liu et al. [15] and subsequent studies by Labrak et al. [16] delve into prompt-based learning and instruction-tuning, presenting promising avenues for efficient domain adaptation. However, these studies reveal an overarching gap: a lack of detailed, practical implementation examples and comprehensive adaptability assessments in varying domain-specific contexts. Recent developments have also seen a rise in innovative approaches like prompt engineering, particularly with generative models like GPT-3 and GPT-4, which have shown potential in adapting to specific NLP tasks without the need for fine tuning the huge LLMs [17]. This move towards efficiency and adaptability is critical in extending the reach and application of NLP technologies, especially in specialized fields like biomedicine.

Despite the numerous attention to this area, the finding from our review of the literature shows very little work has been done that compares the performance of the two different language models in biomedical-domain-specific NER. With this study, we aim to shed more light on this topic by assessing the performance of using the few-shot prompting of an LLM and model adaptation of transformer models to identify and categorize key pieces of information in biomedical texts.

III. Methodology

An overview of the methodology we employ in this study is shown as a flowchart in Figure 1. In general, the method involves data collection and data pre-processing stages, followed by model selection by considering a wide range of transformer-based large language models. The model adaptation stage is then carried out on the chosen models to tailor the model to the chosen domain before performing the NER tasks and evaluating their performances relative to each other.

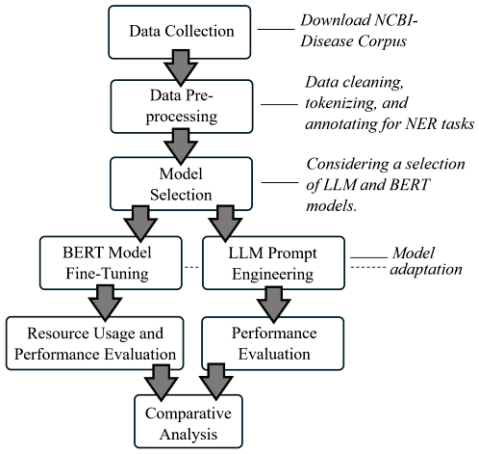

Fig. 1. An overview of the proposed methodology

Data Collection and Pre-processing Stage

In this part, we focused on gathering and preparing the data needed for our biomedical Named Entity Recognition (NER) tasks. We used the NCBI-Disease Corpus [18], a well-known dataset in the biomedical field, which consists of 793 abstracts from PubMed. These abstracts are divided into training, development, and testing sets with detailed annotations for disease mentions. This dataset was chosen because of its

relevance to our study and its widespread use in the biomedical NER community, making it a standard for comparison. The dataset is specifically structured for NER tasks, with annotations that mark the disease and type of disease mentioned within the text. This structure is crucial for our study as it provides a clear framework for training and evaluating our models. The dataset presents text data with annotations embedded within, categorizing text spans into specific biomedical entities such as “SpecificDisease”, “DiseaseClass”, “CompositeMention” and “Modifier”. Each entry is uniquely identified and associated with relevant sections of text, detailing genetic mappings, disease associations, and other biomedical phenomena.

The data pre-processing stage involved formatting and segmenting or tokenizing it into tokens suitable for training the pre-trained language models. Then, we annotated the text according to the BIO tagging scheme, which stands for Beginning, Inside, and Outside. This scheme is standard for NER tasks and involves labeling each word in a sentence to indicate whether it is the beginning of an entity, inside an entity, or outside any entity. The purpose of this meticulous data collection and pre-processing process is twofold. First, it ensures that the models we are testing have a solid foundation of high-quality, well-structured data. Second, it allows us to compare the performance of different models under consistent conditions. By using a standard, recognized dataset and a clear, systematic approach to data preparation, we can ensure that our results are reliable and comparable with other studies in the field.

Model Adaptation Stage

The domain-specific adaptation process is a critical step in adapting pre-trained language models to our specific biomedical NER task. This section outlines how we fine-tuned each selected model to optimize its performance for identifying biomedical entities within text data. Our fine-tuning process began with the standard procedure of adapting each pre-trained model to the NCBI-Disease Corpus. This involved:

1. Data Integration: We integrated our pre-processed biomedical dataset into each model's training framework, ensuring that the format matched the model's requirements. This typically included transforming the text into tokens and aligning these with the corresponding entity labels.
2. Hyperparameter Adjustment: We configured each model's hyperparameters, such as learning rate, batch size, and number of training epochs, based on preliminary tests to find the balance between training time and model performance. The common starting point was a learning rate of 5e-5, a batch size of 32, and three training epochs, adjusting as necessary based on initial results.
3. Training Environment Setup: We ensured that each model was fine-tuned in a controlled environment, typically utilizing GPU acceleration to expedite the training process. This consistency helps in comparing model performances fairly.
4. Model Training: Each model was trained on the annotated training subset of our dataset, using the adjusted hyperparameters. During training, models learned to predict entity labels for text tokens, adapting their parameters to better fit our biomedical NER task.
5. Validation and Adjustment: After initial training, we used the development subset of our dataset to evaluate each model's performance, making further adjustments to hyperparameters if necessary to improve results.

The fine-tuning process was iterative, with adjustments made based on performance metrics and observed challenges. This iterative approach allowed us to refine each model's ability to handle the complexities and nuances of biomedical NER tasks effectively. In summary, the fine-tuning process tailored each pre-trained model to our specific biomedical NER task, with careful adjustments and evaluations ensuring optimal performance. This detailed and methodical approach laid the groundwork for the models' successful application to biomedical text, as discussed in the subsequent results and discussion chapters.

Unlike the transformer-based language models, LLMs are used in a slightly different manner due to their design as a generative model. Instead of traditional fine-tuning, we employed few-shot prompt engineering, creating and refining effective prompts that guided the model to identify and categorize biomedical entities within the text. Prompt engineering involves crafting input prompts that guide the language model to perform specific tasks such as identifying and classifying biomedical entities within text. Unlike fine-tuning, where the model's weights are adjusted, prompt engineering keeps the model's weights fixed and instead modifies the input data to steer the model's output toward the desired task. This approach leverages the generative capabilities of the LLM to interpret and respond to structured prompts, making it suitable for NER tasks without the need for extensive retraining. The design of the prompts considers the following criteria:

1. Prompt Structure: We developed structured prompts that included instructions, context, and examples. Each prompt was designed to clearly convey the task to the model, outlining what biomedical entities are and how they should be identified within the text.
2. Contextual Information: To assist the model in understanding the biomedical context, we included brief descriptions or examples of biomedical entities, ensuring the prompts were anchored in the relevant domain.
3. Task-Specific Instructions: We crafted prompts that explicitly instructed the LLM to identify and label entities within the text. This included using specific tags like B-Disease, I-Disease, and O (Outside) to classify each word or phrase accordingly.
4. Iterative Refinement: Initial prompts were tested and refined based on the model's responses. This iterative process allowed us to fine-tune the wording and structure of the prompts to improve the model's performance on the NER task.

We then evaluated the output and made necessary adjustments to the prompts, seeking to improve clarity and effectiveness. This included rephrasing instructions, adding more contextual information, or providing additional examples.

Performance Evaluation

The effectiveness of each approach in NER tasks was evaluated based on how well the models identified and

classified entities in the test texts. For each type of entity, we use three standard NER evaluation metrics namely, a) Precision (P) which measures the percentage of correctly predicted entities from the total number of entity predictions made, b) Recall (R) which measures the model's ability to identify all relevant entities, and c) F1-score, the harmonic mean of the two calculated as:

$$F_1 = 2 \times \frac{P.R}{P+R} \quad (1)$$

In addition, we also use the Accuracy performance metric when analyzing the overall performance of each model.

IV. Implementation and Experimental Results

The proposed methodology is implemented in Python 3.9 on Jupyter Notebook hosted by Google Colab. Additional libraries include Pandas, Numpy, Transformers, and OpenAI. The training datasets, pre-trained transformer models, and data tokenizer function are obtained via the Hugging Face platform [19]. Five pre-trained models from different variations of the BERT transformer architecture are used. They are the original, or cased, BERT [6], PubMedBERT [5], SciBERT [20], ClinicalBERT [7], and DistilBERT [21]. For the LLM, we consider the latest GPT-4.

The process of fine tuning all BERT models is described in the previous section. While the general approach remained consistent, certain model-specific adjustments were necessary during the implementation. These are:

- BERT, PubMedBERT, SciBERT, ClinicalBERT: These models were fine-tuned with a focus on adapting their understanding of biomedical terminology. For PubMedBERT and SciBERT, less adjustment was needed in terms of learning biomedical terms, given their pre-training on scientific literature.
- DistilBERT: Given its smaller size, we monitored performance closely to ensure that the reduction in parameters did not significantly impact its ability to recognize biomedical entities.

The few-shot prompt approach is in essence a refined version of the zero-shot approach. First, we established a connection to OpenAI's service by using our unique API key. This step is essential for accessing the "gpt-4-turbo-preview" variant of the GPT-4 model. At the time of this study, costs of OpenAI GPT-4 per 1k tokens were approximately $0.001 for input and $0.002 for output. We crafted structured prompts that instruct GPT-4 to identify different biomedical entities within texts. These prompts include explanations, examples, and HTML tagging rules to ensure the model understands the task correctly. The components of the few-shot prompt design that we use are shown in Figure 2 below.

We implemented a function to send text to GPT-4 and receive the processed, entity-tagged text. This function allows us to input biomedical text into our structured prompt, transforming it into a format that the model can understand and respond to accurately. We processed biomedical texts by stripping them of their original BIO tags and inserting them into our prepared prompts. This transformed text was then sent to GPT-4 for entity tagging. Once GPT-4 processed the texts, we collected the outputs, which include HTML-span tagged entities, and saved them for further analysis.

To evaluate the resource efficiency of our implemented models, we tracked the training time, max RAM usage, and average inference time per batch. The result is summarized in Table 1. This data provides insights into the computational efficiency and practicality of each model in resource-constrained environments. Our results show that while the original BERT model requires the least amount of RAM, the model takes almost twice as much time as the fastest model, ClinicalBERT. On the other hand, the two most resource-hungry models, PubMedBERT and SciBERT, have almost identical average inference time to the BERT model.

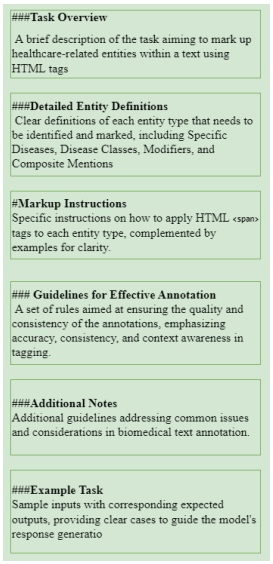

Fig. 2. The components of few-shot prompt engineering design

TABLE I. Resource Usage (BERT models only)

Model	Training time (seconds)	Max RAM used (in MB)	Average Inference Time per batch (in milliseconds)
BERT	4.31	**1084.35**	8.25
ClinicalBERT	**2.60**	1339.20	**4.67**
DistilBERT	**2.71**	1371.82	**4.76**
PubMedBERT	4.39	1779.45	8.37
SciBERT	4.36	1776.93	8.48

The overall NER performance of each BERT model and GPT-4 is shown in Table 2 below. This table provides an at-a-glance comparison of the overall performance of each model, facilitating an understanding of which models are generally more effective for biomedical NER tasks.

TABLE II. Summary of Recognition Performance

Model	Accuracy	Precision (weighted)	Recall (weighted)	F1-Score (weighted)
BERT	0.9475	0.9493	0.9475	0.9468
PubMedBERT	**0.9598**	**0.9618**	**0.9598**	**0.9581**
SciBERT	0.8595	0.8812	0.8595	0.8621
ClinicalBERT	0.9329	0.9323	0.9329	0.9275
DistillBERT	0.9228	0.9300	0.9228	0.9120
GPT-4	0.0640	0.1740	0.0919	0.1203

To delve deeper into each model's capabilities, we also analyze the performance at the entity level. This examination helps to uncover specific strengths and weaknesses of each model concerning the different types of entities identified in biomedical texts. The four entities are CompositeMention (CM), DiseaseClass (DC), Modifier (Mod), and SpecificDisease (SD). The results are shown in Table 3 below.

TABLE III. ENTITY-LEVEL RECOGNITION PERFORMANCE

		CM	DC	Mod	SD
Precision	**BERT**	**0.82**	**0.67**	0.87	**0.82**
	PubMedBERT	0.45	0.66	0.87	0.78
	SciBERT	0.02	0.02	0.87	0.16
	ClinicalBERT	0.67	**0.67**	0.88	0.79
	DistilBERT	0.56	0.42	**0.96**	0.76
	GPT-4	0.00	0.22	0.18	0.17
Recall	**BERT**	**0.36**	0.61	0.74	0.94
	PubMedBERT	0.17	**0.63**	0.50	**0.96**
	SciBERT	0.05	0.16	**0.89**	0.01
	ClinicalBERT	0.22	0.52	0.41	0.91
	DistilBERT	0.06	**0.63**	0.14	0.87
	GPT-4	0.00	0.10	0.17	0.07
F1-Score	**BERT**	**0.50**	0.64	0.80	**0.88**
	PubMedBERT	0.25	**0.65**	0.64	0.86
	SciBERT	0.02	0.03	**0.88**	0.01
	ClinicalBERT	0.33	0.59	0.55	0.84
	DistilBERT	0.10	0.50	0.25	0.81
	GPT-4	0.00	0.14	0.18	0.10

The above table indicates that most BERT models performed relatively well on the SpecificDisease entity, which can be attributed to the distinctive and well-defined nature of disease terms in biomedical literature. SciBERT showed significantly lower performance compared to other models, potentially due to its general scientific corpus training, which may not be as focused on diseases as the biomedical corpora. On the other hand, recognizing the DiseaseClass entity presented challenges across the board, with none of the BERT models reaching the level of performance achieved for SpecificDisease. However, BERT, PubMedBERT, and ClinicalBERT showed comparatively better performance than the rest, indicating their efficacy in grasping broader disease categories likely due to their biomedical contextual training.

On recognizing Modifier entities, which include terms modifying the properties or implications of medical conditions, the table shows a wide range of results. Here, while most BERT models exhibit high precision, their recall performances are often lower and vary significantly with the exception of SciBERT. This suggests this model is better at identifying all relevant modifiers in the text without over-generalizing. Lastly, the table clearly shows that recognizing the CompositeMention entities is the most challenging task, as evidenced by generally lower scores across precision, recall, and F1- score. This indicates a common difficulty in capturing entities that span multiple biomedical concepts, an area that may require more sophisticated approaches or additional training data.

Tables 2 and 3 show clearly the significant challenges faced by the GPT-4 model in adapting to the biomedical NER task through prompt engineering. Overall, the model only achieved 0.0640, 0.1740, 0.0919, and 0.1203 in accuracy, precision, recall, and F1-score, respectively. The low performance highlights the inherent complexities of the biomedical NER task, especially when employing a generalized model like GPT-4 without extensive domain-specific fine-tuning. The low scores across entity types, particularly for 'CompositeMention', underscore the model's difficulty in grasping the distinctions of biomedical terminology and contextual understanding within the constraints of prompt-based learning. These observations emphasize the need for tailored approaches, sophisticated prompt designs, and domain-specific optimizations to leverage the full capabilities of large language models like GPT- 4 in specialized fields such as biomedicine.

V. CONCLUSION

We conclude that, despite being the more popular and the more novel of the two types of language models, the prompt engineering approach with GPT-4 encounters significant challenges, when solving domain-specific terminology comprehension tasks. Its performance markedly lags behind bidirectional transformer-based model approaches, especially using the fine-tuned BERT models, thus underscoring the importance of targeted fine-tuning and domain adaptation. Without it, language models will perform poorly and struggle to meet the minimum standards demanded of them. Our findings also underscore the critical balance between model complexity, accuracy, and resource requirements that need to be considered when choosing which approach and model to use. For future directions, we would argue that enhancing GPT-4's biomedical NER capabilities through refined prompt designs and domain-specific training is necessary, alongside exploring new models and methodologies for resource-efficient biomedical NLP applications.

REFERENCES

[1] J. Chen, Z. Wei, J. Wang, R. Wang, C. Gong, H. Zhang, and D. Miao, "Supplementing domain knowledge to BERT with semi-structured information of documents," *Expert Syst. Appl.*, vol. 235, p. 121054, 2024.

[2] C. Ling, X. Zhao, J. Lu, C. Deng, C. Zheng, J. Wang, T. Chowdhury, Y. Li, H. Cui, X. Zhang, and others, "Domain specialization as the key to make large language models disruptive: A comprehensive survey," *arXiv Prepr. arXiv2305.18703*, 2023.

[3] Y. Ge, W. Hua, K. Mei, J. Tan, S. Xu, Z. Li, Y. Zhang, and others, "Openagi: When LLM meets domain experts," *Adv. Neural Inf. Process. Syst.*, vol. 36, 2024.

[4] 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*, vol. 36, no. 4, pp. 1234–1240, 2020.

[5] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon, "Domain-specific language model pretraining for biomedical natural language processing," *ACM Trans. Comput. Healthc.*, vol. 3, no. 1, pp. 1–23, 2021.

[6] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, no. Mlm, pp. 4171–4186, 2019.

[7] K. Huang, J. Altosaar, and R. Ranganath, "Clinicalbert: Modeling clinical notes and predicting hospital readmission," *arXiv Prepr. arXiv1904.05342*, 2019.

[8] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang, and others, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[9] G. Wang, G. Yang, Z. Du, L. Fan, and X. Li, "ClinicalGPT: large language models finetuned with diverse medical data and comprehensive evaluation," *arXiv Prepr. arXiv2306.09968*, 2023.

[10] Q. Jin, Z. Wang, C. S. Floudas, F. Chen, C. Gong, D. Bracken-Clarke, E. Xue, Y. Yang, J. Sun, and Z. Lu, "Matching patients to clinical trials with large language models," *ArXiv*, 2023.

[11] J. Fries, L. Weber, N. Seelam, G. Altay, D. Datta, S. Garda, S. Kang, R. Su, W. Kusa, S. Cahyawijaya, and others, "Bigbio: A framework for data-centric biomedical natural language processing," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 25792–25806, 2022.

[12] Y. Weng, Z. Wang, H. Liao, S. He, S. Liu, K. Liu, and J. Zhao,

"LMTuner: An user-friendly and highly-integrable Training Framework for fine-tuning Large Language Models," *arXiv Prepr. arXiv2308.10252*, 2023.

[13] Y. Xu, L. Xie, X. Gu, X. Chen, H. Chang, H. Zhang, Z. Chen, X. Zhang, and Q. Tian, "Qa-lora: Quantization-aware low-rank adaptation of large language models," *arXiv Prepr. arXiv2309.14717*, 2023.

[14] G. Pu, A. Jain, J. Yin, and R. Kaplan, "Empirical analysis of the strengths and weaknesses of PEFT techniques for LLMs," *arXiv Prepr. arXiv2304.14999*, 2023.

[15] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Comput. Surv.*, vol. 55, no. 9, pp. 1–35, 2023.

[16] Y. Labrak, M. Rouvier, and R. Dufour, "A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical and biomedical tasks," *arXiv Prepr. arXiv2307.12114*, 2023.

[17] Y. Hu, Q. Chen, J. Du, X. Peng, V. K. Keloth, X. Zuo, Y. Zhou, Z. Li, X. Jiang, Z. Lu, and others, "Improving large language models for clinical named entity recognition via prompt engineering," *J. Am. Med. Informatics Assoc.*, p. ocad259, 2024.

[18] R. I. Do\ugan, R. Leaman, and Z. Lu, "NCBI disease corpus: a resource for disease name recognition and concept normalization," *J. Biomed. Inform.*, vol. 47, pp. 1–10, 2014.

[19] Hugging Face, "Hugging Face. The AI community building the future. The platform where the machine learning community collaborates on models, datasets, and applications." 2023.

[20] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," *arXiv Prepr. arXiv1903.10676*, 2019.

[21] V. Sanh, "DistilBERT, A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter," *arXiv Prepr. arXiv1910.01108*, 2019.