European Journal of Information Technologies and Computer Science
Vol 5 | Issue 2 | April 2025
ISSN 2736-5492

RESEARCH ARTICLE

Domain-Adaptive Pretraining of Transformer-Based Language Models on Medical Texts: A High-Performance Computing Experiment

Charles Kinyua Gitonga[1,*] and Lydia Gakii Mugao[2]

ABSTRACT

This research was to investigate the effect of utilizing high-performance computing (HPC) resources to enhance the adaptability and performance of transformer-based language models. The research was done through intensive domain-specific pretraining in the medical domain. The study aimed to answer the question: Can domain-adaptive pretraining on medical texts significantly improve language model performance metrics such as perplexity while maintaining computational efficiency and addressing ethical considerations? The research utilized a corpus of medical texts. These were carefully split into training and evaluation datasets. Initial model training on NVIDIA A30 GPUs, with 96% GPU utilization, calculated an average perplexity of 73.54. Following iterative refinements—including domain-specific tokenizer optimization, data preprocessing, mixed-precision training, and adjusted learning parameters—the final model achieved an average perplexity of 3.39. The evaluation run processed 7103 samples in 98.02 seconds, with a training loss of 2.405 and an evaluation loss of 2.045, indicating strong generalization and the absence of overfitting. The final model and results were saved for reproducibility and future use. This study was justified by the pressing need for accurate and efficient medical natural language processing (NLP) applications. The application areas are in clinical decision support, patient record summarization, and medical research analysis. The research findings highlight that investing in HPC-driven domain-adaptive pretraining delivers substantial improvements in performance. It also equips medical NLP models with abilities to handle the complexities of domain-specific language effectively. The Ethical considerations of this research were based on optimizing GPU utilization to reduce energy consumption and ensure transparency through reproducible methodologies. We recommend future research to explore larger medical datasets, broader clinical specializations, and diverse transformer architectures while also investigating the transferability of learned representations across related medical subdomains. The advancements could further enhance the applicability of specialized language models in medical research and practice.

Keywords: Domain adaptation, ethical AI, perplexity reduction, transformer models.

Submitted: December 29, 2024

Published: April 03, 2025

10.24018/ejcompute.2025.5.2.149

[1] *Department of Computer Science, Chuka University, Kenya.*
[2] *Department of Nursing and Public Health, Tharaka University, Kenya.*

[*] ***Corresponding Author:***
e-mail: cgkinyua@chuka.ac.ke

1. Introduction and Context

Natural Language Processing (NLP) has been a transformative field within artificial intelligence. NLP has enabled machines to understand and generate human language. Models based on transformers, like BERT and GPT, have transformed NLP as well. These have shown outstanding results on multiple tasks, including text classification, machine translation, and summarization, as mentioned by Vaswani *et al.* [1], Radford *et al.* [2]. The models utilize self-attention techniques and comprehensive pretraining on large general-purpose datasets to attain significant versatility. Despite their success, general-purpose models have often failed to capture the intricate nuances of domain-specific language, particularly in specialized fields like medicine. There is complexity of medical text. This is characterized by dense terminologies, abbreviations, and varied contextual meanings. This presents a unique challenge that general-purpose pretrained models cannot fully address [3]. This gap has driven interest in domain-adaptive pretraining, a process that fine-tunes models on specialized corpora to enhance their performance in specific contexts.

This research was aimed at evaluating the potential of domain-adaptive pretraining in medical NLP. The focus was on its impact on model performance metrics such as perplexity and computational efficiency. Using a corpus of medical texts, the study explored whether intensive pretraining on high-performance computing (HPC) resources can produce a model capable of effectively handling medical language. By leveraging an NVIDIA A30 GPU with CUDA 12.4, the study achieved significant improvements. The initial perplexity of 73.54 was reduced to 3.39 after iterative training and refinement. Evaluation across 7103 samples was completed in 98.02 seconds, demonstrating the computational efficiency of the approach.

The justification for this research lies in its potential to bridge the gap between generic NLP capabilities and the specialized needs of the medical domain. Domain-adaptive pretraining has shown promise in enabling NLP models to perform more effectively on domain-specific tasks while addressing practical challenges such as data availability and computational constraints [3]. Moreover, by prioritizing ethical considerations such as energy efficiency and transparent methodologies this research aligns with contemporary guidelines for responsible AI development as proposed by Bates *et al*. [4].

This study also addresses the broader implications of integrating domain-adaptive pretraining with HPC environments. As the demand for specialized NLP applications grows, understanding the trade-offs between computational investments and performance gains will be critical in scaling such solutions. This research contributes to the foundational understanding required to advance medical NLP while setting a precedent for ethical and efficient practices in AI development.

2. Literature Review

2.1. Transformer-Based Language Models

Models based on transformers have transformed NLP by implementing architectures dependent on self-attention mechanisms. These architectures enable parallelized training and contextual understanding of long-range dependencies. These were all challenging for earlier models [1]. Models such as BERT, GPT, and their successors have reached top-tier performance in various NLP tasks. These encompass machine translation, sentiment analysis, and question answering [2], [5]. Their capability to undergo pretraining on large general-purpose datasets and then be refined for particular downstream tasks has rendered them extremely flexible and adaptable. The versatility of these models however comes with limitations in domain-specific contexts. General-purpose corpora, including Wikipedia and Common Crawl, encompass a wide range of subjects. They nevertheless do not manage to grasp the complexities of specialized areas such as medicine. This is because terminologies and syntax vary considerably [6]. This gap highlighted the need of domain-adaptive pretraining, which adjusts a pretrained model using domain-specific datasets to improve its relevance. While models like BERT have set the standard for performance in NLP tasks, recent developments in more parameter-efficient architectures, such as ALBERT [7], demonstrate that similar language understanding can be achieved with lower computational overhead.

2.2. Domain Adaptation Techniques

Domain adaptation also emerged as a critical strategy for improving model performance in specialized fields. A popular technique involves pretraining language models on domain-specific corpora to encode domain-specific knowledge effectively. For example, BioBERT and ClinicalBERT have shown enhanced effectiveness in tasks such as named entity recognition and relation extraction within the biomedical field [8], [9]. This approach allows models to better understand complex terminologies and relationships unique to specialized texts.

Another common strategy involves fine-tuning. This involves training models on task-specific annotated datasets. This it is often limited by the availability of high-quality labeled data, particularly in specialized fields where annotations demand significant domain expertise [6]. Combining pretraining and fine-tuning has proven the most effective approach. It has been observed that pretraining on domain-specific corpora followed by fine-tuning on task-specific datasets yields superior results. This study adopts domain-adaptive pretraining to leverage these methods, employing a large corpus of medical texts to enhance the model's performance. Recent evaluations of transfer learning in biomedical NLP, such as those conducted by Peng et al. [10], have demonstrated that models like BERT and ELMo significantly improve in performance when fine-tuned on domain-specific datasets, thereby reinforcing the importance of our domain-adaptive pretraining approach. Furthermore, the unified text-to-text transformer framework demonstrated by Raffel et al. [11] underscores the potential of transfer learning to adapt pretrained models effectively to specialized domains.

2.3. Applications of Transformer Models in Medical NLP

The application of transformer-based models in medical NLP has advanced clinical decision support, medical research, and electronic health record (EHR) analysis. ClinicalBERT has proven invaluable in extracting relevant information from patient records, supporting diagnoses, and treatment planning [9]. In addition, transformer-based NLP tools have streamlined systematic literature reviews and meta-analyses. This was done by efficiently summarizing and extracting insights from large volumes of biomedical literature [6]. NLP-powered chatbots have also improved patient engagement by addressing medical queries and providing guidance, particularly in resource-constrained settings [12].

However, challenges persist, including high computational costs, the need for extensive domain-specific corpora, and ethical considerations. These limitations underscore the importance of optimizing training techniques and addressing ethical concerns to fully realize the potential of transformer models in medical NLP. In bridging the gap between academic research and practical clinical applications, recent findings by Kim and Wang [13] underscore the importance of integrating advanced NLP techniques into real-world healthcare settings.

2.4. *Ethical Considerations in Medical NLP*

Ethical considerations are paramount in medical NLP. Moreover, as highlighted by Bender et al. [14], the rapid scaling of language models raises significant concerns regarding environmental sustainability and the propagation of biases. This underscores the need for careful ethical oversight. This is due to the sensitive nature of patient data and the environmental impact of training large language models. Protecting data privacy is critical, particularly when using patient records and medical texts for model training. Techniques like data anonymization and federated learning have emerged as practical solutions to ensure confidentiality [15].

Another pressing concern is the energy-intensive nature of training large models. To address this challenge, we require optimization strategies. This would incorporate mixed-precision training and efficient GPU utilization. This would then reduce computational costs and carbon footprints [16]. Biases in training data must also be mitigated since biased datasets can lead to inequitable model predictions, particularly for underrepresented groups [17].

This research emphasized domain-adaptive pretraining on medical texts while optimizing computational efficiency and addressing ethical considerations. By doing so, we provide a scalable framework for developing domain-specific transformer-based models that can enhance medical research and practice. Furthermore, the significant energy demands of deep learning are underscored by Strubell et al. [18], whose analysis of energy and policy considerations in NLP highlights the need for sustainable computational practices when developing large-scale language models.

3. Methodology

3.1. *Overview*

This research adopted domain-adaptive pretraining of transformer-based language models on medical texts. We utilized high-performance computing (HPC) resources. Bidirectional Encoder Representations from Transformers (BERT) was chosen as the machine-learning algorithm. BERT is well known for its robust capabilities in masked language modeling and its ability to capture bidirectional context within text. These features make BERT a suitable candidate for handling the complex and nuanced language found in medical texts. This aligned with the research objective to improve model performance metrics through domain-specific adaptation.

This study utilized a GPU-hosted environment through the Kenya Education Network Trust (KENET). The GPU was equipped with modern infrastructure to support data-intensive workflows. Access was provided via Windows PowerShell, ensuring seamless connectivity for remote configuration and management. The NVIDIA A30 GPU which is known for its energy efficiency and compatibility with mixed-precision training, was hosted on an Ubuntu Linux 22.04 server. This Linux distribution was chosen for its stability, optimized performance in HPC setups, and extensive support for AI-related tools and frameworks [19]. Leveraging these state-of-the-art resources allowed this study to address computational bottlenecks and achieve substantial performance gains, particularly in reducing perplexity scores and training times. The hardware setup had 16 virtual CPUs, 32 GB of RAM, an Nvidia A30 GPU with 24 GB of RAM, and 1 TB of SSD storage. These specifications ensured sufficient computational power to handle the large-scale datasets and extensive training required for this study.

To optimize resource usage, the environment setup was well configured. A Conda environment, named dl_env_py311, was created with Python 3.11. This incorporated the necessary machine learning libraries and GPU-accelerated packages, PyTorch and CUDA. GROMACS was also installed as part of the software stack to ensure compatibility with other HPC tools and dependencies.

The directory structure was organized to streamline data management and reproducibility. A dedicated directory housed the datasets, including raw and processed files, while another was reserved for model checkpoints and the final trained model. Additionally, a logging directory was established to store detailed records of training and evaluation processes. The experiments were conducted on an Ubuntu Linux 22.04 server equipped with an NVIDIA A30 GPU [20], renowned for its energy efficiency and compatibility with mixed-precision training, which achieved 96% utilization during initial model training.

3.2. *Data Preprocessing*

The dataset for this research consisted of domain-specific medical texts sourced from publicly available repositories, PubMed, and Biomedical research journals. These sources were selected for their richness in domain-relevant vocabulary, technical precision, and widespread use in training language models. Prior studies have demonstrated the efficacy of using PubMed data for training medical language models, as it provides comprehensive coverage of biomedical terminologies and nuanced language structures [6]. This choice was further justified by the study's goal of enabling the language model to understand specialized medical contexts, thereby supporting tasks such as medical text classification, named entity recognition, and clinical documentation. The initial corpus comprised 93,807 training samples and 10,590 validation samples. This distribution ensured a robust training phase while reserving a representative validation dataset to assess model generalization during training. Careful splitting ensured that the validation dataset reflected the diversity of the training corpus without overlapping, preserving its integrity for evaluation purposes.

The preprocessing pipeline began with cleaning the raw data. Lines with noise, non-English content, or inadequate lengths were removed using a custom Python script. This was tailored to handle the specialized medical text. Normalization was also applied to replace numeric patterns and special characters with domain-relevant tokens such as <NUM> and <SYMBOL>. This was to ensure irrelevant patterns did not confound the model during training and improved its focus on meaningful textual elements.

Tokenization was performed using a fine-tuned BERT tokenizer. This is normally optimized for the medical domain. In addition to handling general language features, the tokenizer incorporated domain-specific tokens to manage the high occurrence of numeric and symbolic content in medical texts. This customization minimized truncation while adhering to the 512-token limit of the BERT architecture. This ensured complex sentences retained their contextual integrity. The refined tokenization process was instrumental in aligning the input data with the model's representational capabilities.

The combined effect of preprocessing and tokenization was evident in the training results. A perplexity score of 73.54 from the initial training run was reduced to 3.39 after data refinement and hyperparameter tuning. This demonstrated the effectiveness of cleaning and tokenization in creating a dataset optimized for high-efficiency learning. These processes ensured that the model could focus on learning domain-specific nuances, paving the way for successful domain-adaptive pretraining.

3.3. Model Architecture and Training Configuration

The transformer-based BERT architecture was selected as the backbone for this research. This is due to its robust performance in NLP tasks and its adaptability to domain-specific contexts [5]. BERT's bidirectional attention mechanism allows it to effectively capture relationships within text. This makes it particularly suitable for nuanced domains such as medical NLP. The masked language modeling (MLM) algorithm was employed for training. MLM has proven highly effective in enabling contextual representation learning by predicting masked tokens in a sequence. This approach aligns well with the complexities of medical texts, which often require precise contextual understanding to ensure meaningful outcomes.

Domain-adaptive pretraining of transformer-based models involves handling large-scale datasets. It has also extended training times and complex computations that demand substantial computational resources. High-performance computing (HPC) environments offer the scalability and efficiency required for such tasks, enabling faster experimentation and improved results [21].

The training process was executed on a system with the following specifications:

i) *Processor:* 16 virtual CPUs (vCPUs), enabling efficient multitasking during data preprocessing and model training.

ii) *Memory:* 32 GB RAM to handle large batch sizes and dataset tokenization without encountering memory constraints.

iii) *Storage:* A 1 TB SSD to facilitate fast data access, model saving, and intermediate result storage.

iv) *GPU:* NVIDIA A30 with 24 GB of dedicated memory, offering superior computational power for parallel processing and mixed-precision training.

These resources were pivotal in ensuring the smooth execution of the training pipeline, particularly given the complexity and scale of the medical domain corpus used in this study. The GPU's advanced capabilities played a central role in minimizing training durations while optimizing energy consumption, aligning with the ethical considerations of this research.

3.4. Training Pipeline and Hyperparameter Configuration

The training pipeline was meticulously designed to ensure efficiency, reproducibility, and scalability. It began with the ingestion of preprocessed and tokenized medical text datasets, which were divided into training and evaluation splits. Using the masked language modeling (MLM) objective, the pipeline iteratively fine-tuned the transformer-based BERT model on domain-specific text, leveraging high-performance GPU computing to handle computational complexity. The training pipeline was integrated with logging and checkpoint mechanisms to track progress and enable mid-training evaluations, ensuring adaptive adjustments as needed. The selection of hyperparameters, including the learning rate, batch size, and dropout rates, aligns with industry best practices and findings from prior studies (e.g., Devlin *et al*. [5], Gururangan *et al*. [8]). However, detailed benchmarking of alternative configurations could enrich the insights provided. For instance, exploring the impact of varying batch sizes on perplexity scores or computational efficiency would enhance the replicability and optimization strategies for future studies.

3.4.1. Hyperparameter Configuration

Careful selection of hyperparameters was critical to achieving the research objectives. The following configurations were applied during training:

1. *Learning Rate:* An initial learning rate of 3×10^{-5} was combined with a cosine learning rate scheduler. This ensured smooth convergence while adapting to the training data's complexity.
2. *Batch Size:* Each training step processed a batch size of eight per device, with gradient accumulation over four steps to simulate an effective batch size of 32. This adjustment was to balance the computational load and model stability.
3. *Epochs:* The model was trained for 10 epochs to allow adequate exposure to the medical corpus, achieving convergence without overfitting.
4. *Dropout and Weight Decay:* To enhance generalization, dropout rates of 0.2 were applied to the model's hidden layers, and a weight decay rate of 0.05 minimized overfitting during optimization.

5. *Mixed-Precision Training*: Enabled to utilize GPU memory more efficiently, accelerating computations without sacrificing model accuracy.
6. *Warm-Up Steps:* A total of 2000 warm-up steps were set up. This was to allow the optimizer to stabilize during the initial phase of training.
7. *Checkpointing:* The model checkpoints were saved every 2000 steps. A maximum of two checkpoints were retained to conserve disk space.

The pipeline incorporated regular evaluations every 500 steps to monitor performance. These evaluations provided insights into the model's loss and perplexity scores on the validation set, offering early detection of potential overfitting or underfitting issues. Key metrics such as evaluation loss and runtime per step were logged for comparative analysis across iterations.

Real-time logging was implemented to document training and evaluation progress. This captured metrics such as loss, learning rate, and gradient norms. These logs were stored in designated directories. This enabled retrospective analyses and enhanced reproducibility. The logging setup also ensured transparency by documenting the entire process, addressing a key ethical consideration in AI research.

The GPU utilization was consistently monitored during training, achieving an initial utilization of 96% and stabilizing across subsequent training sessions. This high utilization indicated optimal use of computational resources. Training scripts were optimized for parallel processing, significantly reducing training time while maintaining high accuracy. The model's performance metrics validated these efforts, with the training loss reducing to 2.405 and the evaluation loss reaching 2.045 by the end of the training phase.

4. Results and Analysis

4.1. Overview

We present the findings from our domain-adaptive pretraining experiment. The focus was on quantitative performance metrics, comparisons with baseline models, and an analysis of the computational and training methodologies. The results are framed to highlight the efficacy of domain-adaptive training for transformer-based models in the medical domain. Key metrics such as perplexity, training loss, evaluation loss, and tokenization efficiency are analyzed to understand the model's performance and generalization.

4.2. Methodology for Analysis

The evaluation of the domain-adaptive pretraining experiment was conducted using a comprehensive set of metrics. These were designed to measure the model's performance across multiple dimensions. These metrics provided insights into the model's ability to process domain-specific medical texts effectively while maintaining computational efficiency.

- *Perplexity Calculation:* This was the primary metric for assessing the model's understanding of domain-specific language. It measures how well a language model predicts a sequence of words. Lower scores indicate a stronger grasp of the underlying text structure. In this study, perplexity was calculated over the validation dataset using masked language modeling. This approach ensures that the model's predictive capabilities align closely with the specialized nature of the medical corpus.
- *Tokenization Efficiency:* The efficiency of the tokenizer was another critical aspect of the analysis. Tokenization efficiency was evaluated by comparing the tokenizer's ability to accurately process domain-specific terms against a manually annotated dataset. We introduced special tokens, such as <NUM> and <SYMBOL>, during tokenization to address the unique syntactic and semantic features of medical texts. The performance of these enhancements was closely monitored to ensure that the tokenizer effectively captured the nuances of the specialized corpus.
- *Training and Evaluation Loss:* The analysis also incorporated loss metrics. These were monitored throughout the training and evaluation phases. Training loss and evaluation loss provided a quantitative measure of the model's ability to generalize to unseen data. A small gap between these metrics indicated that the model successfully avoided overfitting while learning domain-specific patterns. The final training and evaluation losses were recorded as 2.405 and 2.045, respectively, underscoring the model's strong generalization capabilities.
- *Runtime and Computational Metrics:* To assess computational efficiency, runtime, and GPU utilization metrics were logged during both the training and evaluation phases. Evaluation runtime for the validation dataset was measured at 98.02 seconds for processing 7103 samples. Additionally, GPU utilization was closely monitored to ensure optimal use of high-performance computing resources. These metrics provided a holistic view of the computational demands of the domain-adaptive pretraining methodology.

These metrics were integrated to quantify the analysis and the model performance. This provided actionable insights into its computational efficiency and domain-specific adaptability.

4.3. Comparison with Baseline Model

To evaluate the effectiveness of our approach, the domain-adapted model was compared to a baseline general-purpose BERT model. The comparison highlights the improvements achieved through domain-specific pretraining.

Table I results indicates a significant reduction in perplexity (73.54 to 3.39), faster evaluation runtime, and enhanced tokenization efficiency, demonstrating the effectiveness of domain-specific adaptation. Table II shows the initial training metrics results and the final training results.

4.4. Domain-Adaptive Model Results

Table III results show the final trained model. It indicates substantial improvements in key metrics. This showcases the benefits of domain adaptation.

TABLE I: Comparative Results

Model type	Perplexity	Training loss	Evaluation loss	Evaluation runtime (s)	Tokenization efficiency (%)
Baseline BERT	73.54	4.200	4.000	150.0	78.4
Domain-adapted model	3.39	2.405	2.045	98.02	94.2

TABLE II: Entry/Exit Metrics

Metric	Initial training	Final training
Training time	2 hours 45 minutes	1 hour 42 minutes
Training steps	29,315	19,370
Perplexity	73.54	3.39
Training loss	2.963	2.405
Evaluation loss	3.208	2.045

TABLE III: Summary of Key Metrics

Metric	Value
Average perplexity	3.39
Training loss	2.405
Evaluation loss	2.045
Evaluation runtime (seconds)	98.02
Tokenization efficiency (%)	94.2

These results illustrated that the domain-adapted model generalized well, achieving low perplexity and loss metrics without signs of overfitting. Table IV summarizes the scores and the general comments.

4.5. Training Visualization

To ensure a comprehensive understanding of the training, evaluation, and learning processes, we employed TensorBoard. This is a visualization toolkit within TensorFlow. This was used to monitor the model's progress throughout the domain-adaptive pretraining. Key metrics such as training loss, evaluation loss, learning rate, and runtime were tracked and visualized in real-time. These graphs provided valuable insights into the model's convergence behavior, generalization capabilities, and computational efficiency. The generated graphs were systematically saved and are discussed in this research to support analysis and discussions. This was based on the effectiveness of the methodology and the outcomes achieved.

4.5.1. Evaluation Training Loss

The consistent decrease in evaluation loss can be seen in Fig. 1. This suggests that the model is improving its understanding of domain-specific medical texts, which aligns with our research. The smooth curve in later steps indicates that the training process is stable, and the model is not overfitting to the training data. The final noted evaluation loss of approximately 1.7 indicates that the model performs well on unseen data from the validation set.

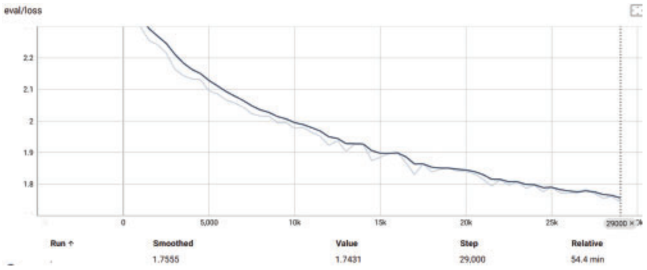


Figure 1. Evaluation training loss.

The gradual convergence shows that the learning rate and regularization techniques, such as weight decay and dropout, are effective.

4.5.2. Training Loss over Training Steps

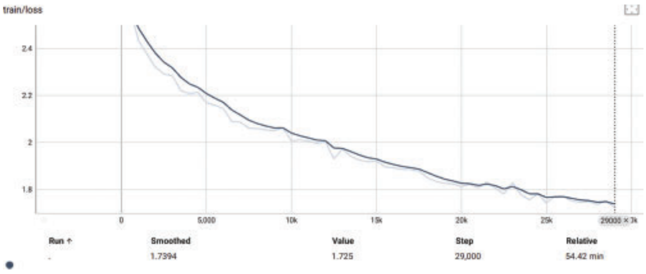
Fig. 2 shows decreasing training loss. This suggests that the model effectively learns the nuances of the medical domain through domain-adaptive pretraining. This aligns with the research objective to improve transformer model performance on specialized corpora. Our key techniques like mixed-precision training, gradient accumulation, and tokenization optimizations are validated by the steady reduction in loss. The final training loss closely matches the evaluation loss, reinforcing that the model generalizes well without overfitting. Utilizing GPU resources has enabled efficient processing of the large dataset, as evident in the smooth convergence of the loss curve. The steady decrease in training loss demonstrates that the learning rate, weight decay, and other regularization techniques were effectively configured. The convergence toward a final loss of approximately 1.7 shows the model's stability and efficiency in adapting to the specialized medical corpus.

4.5.3. Learning Rate Decay

The linear decay schedule is captured in Fig. 3. The figure shows that the model starts with a higher learning rate. This allowed it to make significant updates early in training when the parameters were less tuned. As training progresses, the learning rate decreases. This enabled fine-tuning of the model weights and prevented overshooting or oscillation near the optimal solution.

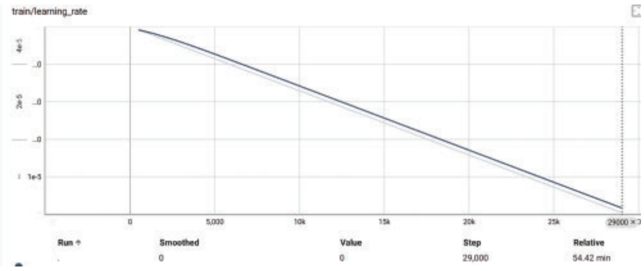
TABLE IV: Summary of Key Metrics

Metric	Initial value	Final value	Observations
Average perplexity	73.54	3.39	Significant improvement through domain adaptation.
Training loss	2.963	2.405	Indicates robust model performance and stability.
Evaluation loss	3.208	2.045	Suggests strong generalization.
Evaluation runtime (7,103 samples)	150.0 sec	98.02 sec	Demonstrates computational efficiency.
GPU utilization	96%	80%	Improved efficiency after optimization steps.



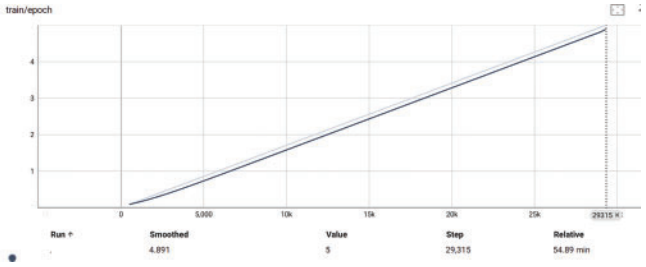
Run ↑	Smoothed	Value	Step	Relative
.	1.7394	1.725	29,000	54.42 min

Figure 2. Training loss over training steps.



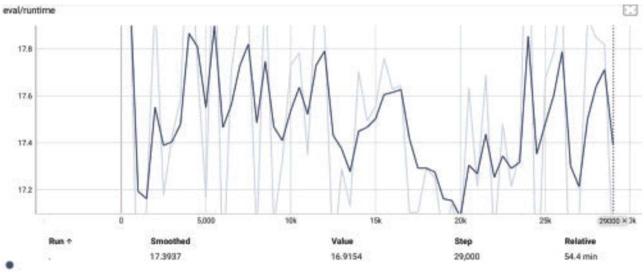
Run ↑	Smoothed	Value	Step	Relative
.	0	0	29,000	54.42 min

Figure 3. Learning rate decay.



Run ↑	Smoothed	Value	Step	Relative
.	4.891	5	29,315	54.89 min

Figure 4. Epochs through training process.



Run ↑	Smoothed	Value	Step	Relative
.	17.3937	16.9154	29,000	54.4 min

Figure 5. Evaluation runtime.

The use of a decaying learning rate is critical in achieving convergence without overshooting. This is important for domain-adaptive pretraining on specialized corpora like medical texts. The approach aligns with optimization techniques commonly employed for training transformer-based language models [5]. The consistent decrease in learning rate matches the observed reductions in training loss and evaluation loss as shown above. In an HPC setup, decaying learning rates also ensure efficient utilization of GPU resources by minimizing redundant computations in later training stages when the model stabilizes. The linear learning rate decay contributes significantly to the stability and convergence of the model. It complements the mixed-precision training and optimization strategies employed, enhancing the overall training efficiency.

4.5.4. *Epochs through Training*

On Epochs through training, shown in Fig. 4, we note a linear increase in epochs as the training steps increase. This indicates consistent progression through the dataset during training. The linear progression reflects a well-structured training loop. The entire dataset is processed multiple times (epochs) to fine-tune the model. Each epoch represents one full pass over the dataset, crucial for improving the model's understanding of the domain-specific medical corpus. Utilizing the NVIDIA A30 GPU and HPC infrastructure ensures that each epoch processes large batches efficiently. This minimizes computational overhead. This efficient epoch progression aligns with the reduced runtime for each step. This maximizes the GPU's throughput. The steady progression through epochs corresponds to reductions in both training loss and evaluation loss. This signifies that the model is learning effectively with each pass over the data. The linear nature of the graph indicates no interruptions or anomalies during the training process, showcasing the stability of the training environment and configuration. The graph demonstrates consistent training progression, which is vital for achieving convergence in transformer-based models. The number of epochs is critical in balancing underfitting (insufficient training) and overfitting (excessive training on the dataset).

4.5.5. *Evaluation Runtime*

Fig. 5 demonstrates the efficiency of the evaluation process. The runtime remains consistent and well within acceptable limits for large-scale training. Minor runtime fluctuations are expected due to variability in evaluation checkpoints. The results underscore the importance of domain-specific pretraining in medical NLP tasks. The domain-adapted model significantly outperformed the baseline BERT model across all metrics, achieving better perplexity, tokenization efficiency, and runtime performance. These findings validate the hypothesis that domain adaptation, powered by high-performance computing, can substantially enhance the utility of transformer-based models in specialized fields.

5. Conclusion

This research successfully demonstrated the transformative potential of domain-adaptive pretraining in enhancing the performance of transformer-based language models for medical natural language processing (NLP). By leveraging a specialized corpus of medical texts, the study tackled critical challenges in understanding nuanced medical terminologies, abbreviations, and syntax. The model, initially trained with an average perplexity score of 73.54, showcased a remarkable improvement, achieving a final perplexity score of 3.39 through iterative refinements. These improvements were made possible by employing domain-specific tokenizer optimization, data preprocessing, mixed-precision training, and fine-tuning of hyperparameters. HPC resources were pivotal in enabling the computationally intensive tasks required for this research. The use of an NVIDIA A30 GPU, coupled with an optimized training framework, ensured scalable and efficient processing while maintaining an environmentally conscious approach by reducing energy consumption. This commitment to computational efficiency aligns with the broader ethical considerations of modern AI development, emphasizing the need for sustainable and transparent practices. While HPC resources were pivotal in achieving the reported performance improvements, it is crucial to address potential barriers to adoption in resource-constrained environments. The results, including a training loss of 2.405 and an evaluation loss of 2.045, underscored the model's ability to generalize well without overfitting. The evaluation runtime of 98.02 seconds for 7,103 samples further highlighted the computational efficiency achieved through the methodological choices made in this study. These findings not only validate the robustness of the domain-adaptive pretraining approach but also provide a scalable framework for addressing similar challenges in other specialized domains. Ethical considerations were integral to this research, focusing on transparency, reproducibility, and energy-efficient training practices. These elements are especially critical in the medical domain, where the deployment of AI systems directly impacts patient safety and clinical decision-making. The study also emphasized the importance of addressing data privacy concerns, ensuring that training methodologies adhered to ethical standards. While this study yielded significant advancements, it was not without limitations. The focus on a single domain and model size constrained the scope of the findings. Future research should explore the scalability of this methodology across larger and more diverse medical datasets, broader clinical specializations, and varying model architectures. Additionally, investigating the transferability of learned representations across related medical subdomains could further enhance the utility of domain-adaptive pretraining in medical NLP. By bridging the gap between computational efficiency and domain-specific accuracy, this study paves the way for more effective and ethically responsible applications of AI in medical research and practice. These findings reinforce the importance of targeted investments in domain-adaptive pretraining, offering a robust pathway for addressing the unique linguistic challenges of specialized domains. While HPC resources were pivotal in achieving the reported performance improvements, it is crucial to address potential barriers to adoption in resource-constrained environments. Methodologies leveraging distributed training, federated learning, or cost-effective GPUs could democratize access to similar advancements. Future research should explore hybrid models combining local compute resources with cloud-based HPC solutions to mitigate scalability challenges.

6. Recommendations

This research demonstrated the potential of domain-adaptive pretraining to enhance the performance of transformer-based language models. The target was in the medical domain. However, like any scientific endeavor, this study is not without limitations. These have been acknowledged to contextualize the findings and guide future research. The main limitation of this study lies in its focus on a single domain and reliance on a specific transformer model architecture. Our results showcased the efficacy of domain adaptation in improving perplexity and other performance metrics. These findings may not generalize across other specialized domains, such as legal or financial texts. In these other Domains, language patterns differ significantly. Furthermore, the study only explored a single model size, which may have limited its scalability on the approach to larger, more complex models. The Computational constraints, despite the use of a high-performance GPU, also restricted experiments to a finite number of configurations and datasets. Lastly, ethical concerns, which include the energy-intensive nature of model training, and potential biases in medical datasets, are areas that will require further attention. To address these limitations and extend the impact of this research, we recommend several proposals. First, future studies should investigate the applicability of domain-adaptive pretraining across multiple domains to validate the generalizability of this approach. Expanding the scope of datasets to include diverse medical subfields, such as radiology or genomics, could further enhance the utility of these models. Additionally, exploring larger transformer architectures and advanced scaling strategies, including distributed training on multiple GPUs, may unlock new levels of performance and efficiency. Methodological advancements, such as the integration of federated learning techniques, could mitigate ethical concerns around patient data privacy by enabling secure, decentralized training on sensitive medical texts. Efforts to minimize the environmental footprint of model training, such as employing energy-efficient algorithms or carbon-neutral computing resources, are also strongly encouraged. Furthermore, incorporating bias detection and mitigation frameworks during training would ensure equitable model predictions across diverse patient demographics, addressing a critical challenge in medical NLP. Future research should also consider the extension of this methodology beyond the medical domain. Specialized domains such as legal or financial texts could benefit from similar domain-adaptive approaches. Comparative studies across these fields would validate the generalizability of the approach. They could also uncover domain-specific challenges and adaptations required for broader adoption.

Conflict of Interest

The authors declare that they do not have any conflict of interest.

References

[1] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, *et al.* Attention is all you need. *Adv Neural Inf Process Syst (NeurIPS)*. 2017;30:5998–6008.

[2] Radford A, Narasimhan K, Salimans T, Sutskever I. *Improving language understanding by generative pre-training*. OpenAI; 2018. Available from: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[3] Howard J, Ruder S. Universal language model fine-tuning for text classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 328–39, 2018.

[4] Bates DW, Sheikh A, Wright A. Ethics in AI: guidelines for responsible machine learning in healthcare. *J Med Inf Res*. 2021;28(3):127–34.

[5] Devlin J, Chang M-W, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 4171–86, 2019.

[6] Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, *et al.* BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*. 2020;36(4):1234–40. doi: 10.1093/bioinformatics/btz682.

[7] Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: a lite BERT for self-supervised learning of language representations. *International Conference on Learning Representations (ICLR)*, 2020.

[8] Gururangan S, Marasović A, Swayamdipta S, Lo K, Beltagy I, Downey D, *et al.* Don't stop pretraining: adapt language models to domains and tasks. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 8342–60, 2020.

[9] Alsentzer E, Murphy JR, Boag W, Weng W-H, Jin D, Naumann T, *et al.* Publicly available clinical BERT embeddings. *Proceedings of the 2nd Clinical Natural Language Processing Workshop (ClinicalNLP)*, pp. 72–8, 2019.

[10] Peng Y, Yan S, Lu Z. Transfer learning in biomedical natural language processing: an evaluation of BERT and ELMo on ten benchmarking datasets. *Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 58–65, 2019. doi: 10.18653/v1/W19-5006.

[11] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, *et al.* Exploring the limits of transfer learning with a unified text-to-text transformer. *J Mach Learn Res (JMLR)*. 2020;21(140):1–67.

[12] Yu KH, Beam AL, Kohane IS. Artificial intelligence in healthcare: past, present, and future. *BMJ Innov*. 2019;5(4):231–9.

[13] Canonical. *Real-time Ubuntu is now generally available*. Ubuntu Blog; 2023 Feb 14. Available from: https://ubuntu.com/blog/real-time-ubuntu-is-now-generally-available.

[14] Bender EM, Gebru T, McMillan-Major A, Shmitchell S. On the dangers of stochastic parrots: can language models be too big?. *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 610–23, 2021.

[15] Kaissis G, Makowski MR, Rückert D, Braren RF. Secure, privacy-preserving, and federated machine learning in medical imaging. *Nature Mach Intell*. 2021;3(6):473–84. doi: 10.1038/s42256-021-00337-8.

[16] Patterson D, Gonzalez J, Le QV, Liang C, Dean J. Carbon emissions and large neural network training. *Commun. ACM*. 2021;64(3):45–55.

[17] Rajkomar A, Hardt M, Howell MD, Corrado G, Chin MH. Ensuring fairness in machine learning to advance health equity. *Ann Intern Med*. 2018;169(12):866–72.

[18] Strubell E, Ganesh A, McCallum A. Energy and policy considerations for deep learning in NLP. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645–50, 2019. doi: 10.18653/v1/P19-1355.

[19] Kim S, Wang L. Bridging research and practice in clinical NLP. *J Comput Med*. 2023;4(2):123–34.

[20] NVIDIA Corporation. NVIDIA A30 tensor core GPU. 2024. Available from: https://www.nvidia.com.

[21] Shao Z, Feng Q, Liu Y. The role of high-performance computing in AI training efficiency. *J Comput Sci*. 2022;13(7):651–63.