SYNPO: Synthetic Data Augmentation Through Iterative Preference Optimization on Large Language Models for Clinical Problem Summarization

Shayan Chowdhury SCHOWDHURY7@BWH.HARVARD.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA
Columbia University, NY, USA

Shan Chen SCHEN73@BWH.HARVARD.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA

Ciel Wang YW7104@NYU.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA
New York University, NY, USA

Luoyu Zhang LZHANG92@BWH.HARVARD.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA

Jack Gallifant JGALLIFANT@BWH.HARVARD.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA

Danielle Bitterman DBITTERMAN@BWH.HARVARD.EDU
Artificial Intelligence in Medicine Program, Mass General Brigham, Harvard Medical School, Boston, MA, USA

Abstract

High-quality annotated datasets are often limited in the domain of clinical health data, posing challenges for training effective machine learning models to solve clinical natural language processing (NLP) tasks. This paper introduces SYNPO, a method leveraging large language models to iteratively enhance synthetic annotated data generation. Specifically, in the clinical Problem List Summarization task, we demonstrate that models fine-tuned on synthetic data generated by SYNPO from as low as 10 gold examples outperform those trained on the full gold dataset of 595—with performance peaking at 100 gold examples, suggesting an optimal balance between data quantity and model generalization. We validated SYNPO-generated summaries through expert clinician evaluations, revealing both strengths and limitations. While promising, SYNPO requires further validation across tasks and datasets to ensure broader applicability. Future efforts will aim to reduce reliance on privacy-sensitive data, enhance efficiency, and assess generalizability in diverse clinical NLP tasks.

Keywords: Large Language Models, Healthcare, Synthetic Data, Preference Optimization, Clinical Evaluation

Data and Code Availability We focus on the problem list summarization (ProbSum) NLP task from DR.BENCH (Gao et al., 2023), which is a derivative of the Medical Information Mart for Intensive Care III (MIMIC-III) dataset (Johnson et al., 2016). This data is available on PhysioNet, subject to the completion of appropriate data use agreements. Our code is anonymized and provided as supplemental material at https://anonymous.4open.science/r/fake2real/.

Institutional Review Board (IRB) This research was deemed to be non-human subjects research by the Mass General Brigham IRB.

1. Introduction

Natural Language Processing (NLP) in the clinical domain presents unique challenges due to the specialized nature of medical terminology, the complexity of clinical narratives, and the scarcity of large-scale, high-quality annotated datasets (Chapman et al.,

2011; Guevara et al., 2023). These challenges often limit the performance of machine learning models in tasks such as clinical text summarization, entity recognition, and information extraction (Yu et al., 2024; Fan et al., 2024; Wang et al., 2023). Moreover, the development of robust clinical NLP methods is further hindered by traditional annotation processes' time-consuming and expensive nature, which frequently require expert medical knowledge (Tang et al., 2023; Chen et al., 2024b). Significant constraints surrounding patient privacy and data governance add another layer of complexity, complicating the sharing of large clinical datasets and making it difficult to scale, develop, and validate generalizable models (Walonoski et al., 2017; Melamud and Shivade, 2019). These factors collectively contribute to significant research and benchmark development constraints in clinical NLP.

Synthetic data generation has gained attention as a potential solution in response to these challenges. For example, generative adversarial network (GAN)-based methods have shown promise in generating synthetic image data in the medical realm (Frid-Adar et al., 2018). However, semantic consistency and grammatical accuracy have limited existing NLP applications, such as paraphrasing or word swapping to generate synthetic text. In contrast, recent advancements in large language models (LLMs) have opened new possibilities in this space. LLMs are particularly adept at generating fluent, high-quality text, making them well-suited for creating synthetic clinical datasets that can supplement real-world data (Tang et al., 2023; Chen et al., 2024a; Li et al., 2023b).

We introduce a novel approach to generating synthetic annotated clinical text using LLMs, aimed at enhancing clinical NLP models. By supplementing **as few as 10 expert-annotated instances** with generated synthetic data, we reduce the need for extensive manual annotation and accelerate model development. Our method leverages Hong et al. (2024)'s Odds Ratio Preference Optimization (ORPO) algorithm to iteratively improve LLM-generated data, enhancing the performance of downstream models.

In this study, we focus on clinical text summarization, demonstrating how preference learning can be used to improve the generation of synthetic clinical narratives and their corresponding summaries. Our approach to improving performance on this task through synthetic data generation and model fine-tuning follows an iterative process that combines in-context learning (ICL), synthetic data generation, and preference optimization techniques to progressively enhance the quality of generated clinical data and improve model performance.

2. Background

2.1. Prior Work

This work builds on Chen et al. (2024a), which demonstrated that synthetic data could effectively enhance NLP models for clinical tasks by replacing or augmenting gold-standard datasets. The current study extends this by introducing preference learning to iteratively improve the quality of LLM-generated synthetic data, further boosting model performance with **a fraction of gold-labeled examples needed**.

2.2. Synthetic Data Generation & Iterative Improvement

Several papers have tackled synthetic data generation in the general LLM domain (Puri et al., 2020; Li et al., 2023c). Other work involving improving LLM outputs through preference optimization includes Chen et al. (2024c)'s SPIN algorithm, which involves a self-play mechanism where the LLM refines its capability by playing against instances of itself, progressively elevating the LLM's performance and outperforming models trained through direct preference optimization (DPO) (Rafailov et al., 2024) with extra GPT-4 preference data. Pang et al. (2024) demonstrates using DPO to iteratively optimize choosing the best LLM outputs generated using Chain-of-Thought (CoT) reasoning. Lee et al. (2024) demonstrates a *teacher LLM* that iteratively identifies points where a *student LLM* struggles and generates synthetic data to *teach* the latter, targeting its weaknesses. Adler et al. (2024) utilizes synthetic data generation to create over 98% of the data used in training NVIDIA's Nemotron-4-340B-Instruct model, leveraging an iterative "weak-to-strong" approach to refine the synthetic data using increasingly capable models, similar to our own method.

2.3. Problem List Summarization Task

The specific task being assessed in this study is clinical problem list summarization, which involves generating a concise summary of a patient's medical problems based on a longer clinical narrative, derived from Task 6 of the DR.BENCH benchmark (Gao et al.,

2023). The task requires the model to generate a concise and coherent summary that captures key medical problems mentioned in the narratives containing complex specialized medical terminology, making it particularly challenging. Several papers have explored this task: Li et al. (2023a) explored fine-tuning an encoder-decoder language model, while Gao et al. (2024) linked an in-domain knowledge graph with GPT processing to improve performance. Park et al. (2024) explored in-context learning with up to 16 shots on GPT-4.

3. Methods

3.1. Data Preparation

As mentioned in the *Data and Code Availability* section above, we focus on the ProbSum task dataset from DR.BENCH (Gao et al., 2023), with the goal of generating concise summaries from detailed clinical notes. It was developed using MIMIC-III (critical care notes from the Beth Israel Deaconess Medical Center between 2001 and 2012) (Johnson et al., 2016). The dataset consists of training ($n = 595$), evaluation ($n = 75$), and testing ($n = 87$) samples. While the full dataset provides a comprehensive overview of patient conditions into four sections: *Subjective* (patient's description), *Objective* (medical professional's observations), *Assessment* (diagnosis), and *Plan* (treatment plan), we only use the *Assessment* and *Plan* for generating summaries as prior work also has (Gao et al., 2023; Li et al., 2023a). We do standard data preprocessing, which involves removing rows with missing values, followed by basic string cleaning, including trailing whitespaces, newlines, and punctuation.

3.2. Model Architectures

Due to the sensitive nature of the medical data being worked with, we implemented open-source models that could be run and trained locally. For the purposes of this paper, we employed the instruction-tuned Mistral-7B-v0.3 (Jiang et al., 2023) for its robust performance in understanding and generating text data, even without clinical domain-specific pre-training. We implemented the models using the `unsloth` library (Han et al., 2023), allowing for LoRA (Low-Rank Adaptation) for faster training and inference and reduced VRAM usage (Hu et al., 2022). Our detailed experimental setup can be found in Appendix A.6.

3.3. Baseline & Gold Fine-Tuned Performance Assessment

The overall pipeline can be found in Algorithm 1. In the first two steps of the *Baseline* section, we preprocessed our data and then established a performance baseline on the base LLM (LLM_0), conducting zero- and few-shot inference using our gold standard test set ($n = 87$). To establish a baseline for a model fine-tuned entirely on gold-labeled data (in comparison to using synthetic data), we performed supervised fine-tuning (SFT) on the base LLM (resulting in LLM_{gold}) using the Stanford Alpaca dataset prompt template (Taori et al., 2023), which aligns instructions, inputs, and outputs for improved task performance.

3.4. Synthetic Data Generation

Now begins our main method for SYNPO, detailed in Figure 1. First, we prompt the base LLM to generate an initial set of synthetic clinical data, providing 5 in-context examples randomly sampled from the training data to mimic the styling, syntax, and terminology used in real clinical data. Multiple retry attempts were made to address parsing failures in LLM outputs, and duplicates were removed to ensure dataset diversity. The prompt used for this step can be found in Appendix A.4. This is also illustrated as Step 1 of Figure 1 and our *Proposed Method* in Algorithm 1.

3.5. Iterative Preference Optimization Setup

With our first set of synthetic data generated from the base LLM, we employed the ORPO algorithm (Hong et al., 2024) to enhance the quality of the synthetic data through fine-tuning, as seen in Step 2 of Figure 1. ORPO directly penalizes undesirable outputs and reinforces preferred ones using a log odds ratio, without requiring a reference model or warm-up phase.

To construct this dataset, we matched each synthetic instance (*rejected*) with a corresponding gold instance (*accepted*) at random. Each data point in this dataset therefore contained an instruction, input prompt, real example, and synthetic example, which can be found in Appendix A.5.

Using this paired dataset, the first iteration of the SYNPO model for synthetic data generation was fine-tuned using ORPO, yielding $\text{LLM}_{\text{SYNPO}_1}$. The prompt used for this step can be found in Appendix A.5.

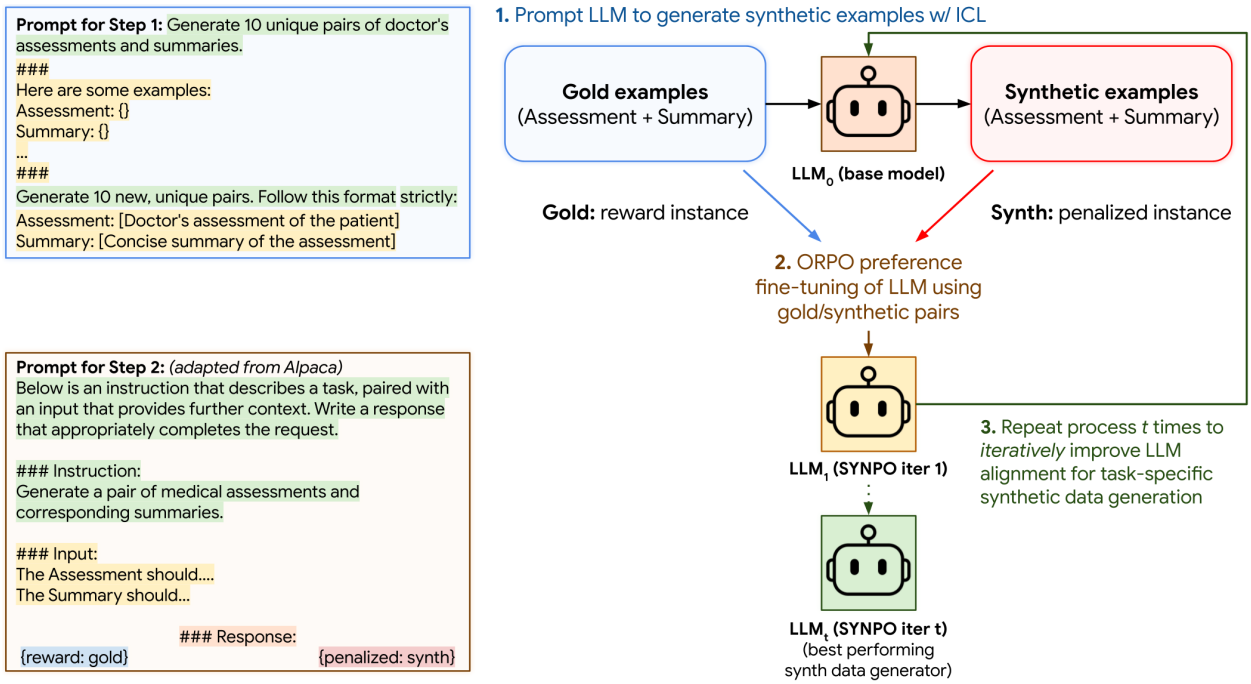

Figure 1: Pipeline for iterative fine-tuning with SYNPO, where a synthetic data generator LLM is refined through repeated preference-based fine-tuning. Full prompts can be found in Appendix A.

3.6. Fine-Tuning on Synthetic Data

With a new batch of synthetic data generated from our new SYNPO model, a new instance of the base LLM (LLM_{synth_1}) was fine-tuned for the ProbSum task. Before the data are fed into the model, we again format the data using the Alpaca dataset prompt template in Appendix A.2, aligning with the model's instruction template and ensuring consistency during training. The fine-tuning process is then carried out while applying LoRA as described in Section 3.2, which updates only a small subset (1-10%) of the model's parameters, significantly reducing computational costs while maintaining effectiveness in alignment (Hu et al., 2022).

Furthermore, several parameters were manually configured through trial and error, including learning rates, weight decay, gradient norms, early stopping, and a cosine learning rate scheduler, as described in greater detail in Section 3.2. The model is evaluated regularly during training on the evaluation set (not the held-out test set), with the best-performing version saved for the next iteration.

3.7. Iterative Refinement

The model fine-tuning, synthetic data generation, and fine-tuning processes were repeated **iteratively** to improve model performance, as seen in step 5 of our *Proposed Method* in Algorithm 1. In each iteration, we fine-tuned a successive SYNPO model to generate better quality synthetic data (LLM_{SYNPO_t}), generated a new batch of synthetic data ($n = 595$), and then used that data to fine-tune the newest iteration of the LLM (LLM_{synth_t}) for predictions. After each iteration, the synthetic model was evaluated using k-shot inference, and the SYNPO model was further trained to refine the quality of synthetic data—and so on. This process was repeated for t iterations until no significant performance gains on the validation set were observed.

3.8. Experimental Setup for Varied Subsets of Gold Data

Experiments were conducted with varying sizes of the real dataset to investigate the impact of the amount of gold-labeled data used in our method (and to

Algorithm 1: SYNPO Pipeline

Symbol	Definition
D_{gold}	Gold dataset
N	Number of examples from D_{gold} used for training, inference, and synthetic data generation
T	Number of SYNPO iterations
LLM_0	Base LLM (no fine-tuning)
LLM_{gold}	LLM fine-tuned only on D_{gold}
$\text{LLM}_{\text{SYNPO}_t}$	LLM to generate improved synthetic data (assessment/summary pairs) at SYNPO iter t
D_{synth_t}	Improved synthetic data generated by $\text{LLM}_{\text{SYNPO}_t}$
LLM_{synth_t}	LLM fine-tuned on D_{synth_t} to generate summaries given an assessment (inference) at SYNPO iter t
k	Number of shots (in-context examples) for testing

Baseline:

1. Preprocess D_{gold}, select N samples $\rightarrow D_{train}$
2. Perform k-shot inference on LLM_0 using D_{test}
3. Fine-tune LLM_0 on $D_{train} \rightarrow \text{LLM}_{gold}$
4. Test LLM_{gold} with k-shot inference on D_{test}

Proposed Method:

1. Generate synthetic data D_{synth_0} using LLM_0
2. Fine-tune LLM_0 on $D_{synth_0} \rightarrow \text{LLM}_{synth_0}$
3. Test LLM_{synth_0} with k-shot inference on D_{test}
4. Train SYNPO model $\text{LLM}_{\text{SYNPO}_1}$ to improve synthetic data generation, optimizing for data similarity to D_{train} and dissimilarity to D_{synth_0}
5. For $t = 1$ to T (or until optimal performance):
   (a) Generate D_{synth_t} using $\text{LLM}_{\text{SYNPO}_t}$
   (b) Fine-tune $\text{LLM}_{synth_{t-1}}$ on $D_{synth_t} \rightarrow \text{LLM}_{synth_t}$
   (c) Test LLM_{synth_t} with k-shot inference on D_{test}
   (d) Train $\text{LLM}_{\text{SYNPO}_{t+1}}$ by optimizing for data similarity to D_{gold} and dissimilarity to D_{synth_t}
6. Outputs: LLM_{synth_T}, $\text{LLM}_{\text{SYNPO}_T}$, D_{synth_t}

demonstrate how the models can perform with varying amounts of gold-labeled data availability). In our work here, we experimented with sampling only 10, 50, 100, and 595 (the full gold dataset size) examples from the gold training dataset for use in inference, synthetic data generation, and SYNPO fine-tuning.

4. Evaluation

4.1. Quantitative Evaluation

During the training process, the models were evaluated on the evaluation set at regular 50-step intervals, with the best-performing checkpoint based on validation loss saved for final use. The resulting fine-tuned models were assessed on the held-out gold standard test set, applying k-shot inference (zero-shot and few-shot, specifically using 2 and 5 randomly selected examples from our training dataset) to determine its performance improvement over the baseline, and to use as a benchmark for evaluating subsequent improvements from SYNPO fine-tuning. The performance metrics measured included ROUGE-L (Lin, 2004) and METEOR score (Banerjee and Lavie, 2005). For reproducibility, we repeated each experiment three times using the same conditions, hyperparameters, and random seeds, and reported the mean and standard deviation of the results in Section 5.

4.2. Qualitative Evaluation

To complement the quantitative evaluation, we also conducted a qualitative analysis of our model's generated summaries with two physicians and two laypeople. Specifically, we randomly sampled 20 examples from the test set and conducted two tests:

1. Comparing summaries generated by the best performing model fine-tuned on all the gold-standard data to the gold-standard summaries
2. Comparing summaries generated by the best performing synthetic model fine-tuned on synthetic data generated using only 10 gold examples to the gold-standard summaries

The goal of this evaluation was to assess whether the model fine-tuned on synthetic data could generate summaries comparable in quality to those generated by the model fine-tuned on gold-standard data. The evaluation interface was built using the Streamlit web application, and participants were asked to select the

summary that best captured the key medical problems mentioned in the assessment portion of the narrative.

We implemented a structured experimental setup to ensure a standardized and reproducible evaluation process. We used two random seeds for each participant to generate two sets of 20 examples each. The first set compared gold-standard summaries with summaries from the model fine-tuned on all gold data. The second set compared gold-standard summaries with those from the synthetic model trained on ten gold examples.

We paired participants (one physician with one layperson) to use the same seeds for their respective tests. This pairing strategy allowed us to control for any potential variations in the randomly selected examples, ensuring that any differences in judgments between experts and non-experts could be attributed to their level of expertise rather than differences in the presented examples.

5. Results

5.1. Iterative Improvements Through SYNPO

Table 2 illustrates the performance trajectory of models trained on synthetic data generated using ten real examples, over five SYNPO iterations. As mentioned in Section 4, we tested all inference strategies (including 0, 2, and 5-shot) and reported the best-performing models. The base LLM achieved a ROUGE-L F1 (RL-F1) score of 21.38. Fine-tuning on the limited gold data ($n = 10$) resulted in a slight performance decrease (RL-F1: 21.16), likely due to the small sample size. However, fine-tuning on the initial synthetic data generated (iteration 0) improved performance overall, with RL-F1 increasing to 23.02.

Subsequent SYNPO iterations demonstrated a general initial trend of performance enhancement. RL-F1 scores improved from 24.38 in iteration 1 to 25.74 in iteration 2. A slight dip occurred in iteration 3 (24.94), followed by recovery and further improvement in iterations 4 and 5, culminating in an RL-F1 of 25.55.

Notably, we observed a trade-off between precision and recall across iterations. While RL-Precision improved from 20.40 (base model) to 36.87 (iter 5), RL-Recall declined from 32.02 to 23.36. This trade-off is discussed further in Section 6.

5.2. Performance on Varied Subsets of Gold Data

To assess the impact of gold data quantity in the SYNPO process, we conducted experiments using varying amounts of gold data, as described in Section 3.8. Figure 2 illustrates the performance of the base language model (Base LLM), the model fine-tuned on the full gold dataset (Gold LLM), and models fine-tuned on synthetic data generated from different numbers of gold examples (Synth LLM).

Given 5 in-context examples, the base LLM exhibited the lowest performance, with an RL-F1 of 20.02 $\pm$ 0.85 (mean $\pm$ std. dev. across 3 experiments using the same conditions). All fine-tuned models, including those trained on synthetic data from as few as 10 gold examples, outperformed the base model.

The model fine-tuned on the full gold dataset (595 examples) achieved an RL-F1 of 23.74 $\pm$ 0.51. However, models fine-tuned on synthetic data consistently outperformed this gold-standard model as well, across all metrics. The synthetic model trained using only 10 gold examples achieved a score of 25.87 $\pm$ 0.48, surpassing the full gold model by over 2 percentage points. We observed a marginal increase in performance with more gold examples, reaching 26.05 $\pm$ 0.88 with 50 examples and peaking at 26.41 $\pm$ 0.89 with 100 examples.

Interestingly, the synthetic model using all 595 gold examples showed a slight decrease in performance (25.24 $\pm$ 1.48) compared to models using fewer examples, though it still outperformed the full gold model. This is also evident in the METEOR and BERTScore-F1 scores, suggesting a potential plateau or overfitting when using larger amounts of gold data in our method.

5.3. Qualitative Analysis

The results from Section 4.2 illustrate that when comparing gold-standard summaries against those produced by the gold-standard model, participants showed a slight overall preference for the gold-standard summaries. The average accuracy in correctly identifying the gold-standard summary was 53.75% (43/80 total judgments). However, individual preference varied, with accuracies ranging from 35% to 65% across participants.

When comparing gold-standard summaries against those generated by the synthetic model, participants demonstrated a marginal preference for the synthetically generated summaries. In this comparison, the

Table 1: Performance of selected models trained on synthetic data generated using only 10 real examples after 5 SYNPO iterations, evaluated using ROUGE-L (RL), METEOR Score, and BERTScore. Numbers in parentheses are the increase/decrease compared to the previous row. Prior baselines from the literature are in Table 3.

Model	RL-F1	RL-Precision	RL-Recall	METEOR	BERTScore-F1
Base LLM	21.38	20.40	32.02	27.79	62.98
Gold LLM	21.16 (–0.22)	19.63 (–0.77)	31.33 (–0.69)	26.49 (–1.30)	62.55 (–0.43)
Synth LLM (iter 0)	23.02 (+1.86)	24.08 (+4.45)	27.51 (–3.82)	25.89 (–0.61)	63.51 (+0.96)
Synth LLM (iter 1)	24.38 (+1.36)	33.03 (+8.95)	23.13 (–4.38)	22.61 (–3.28)	64.26 (+0.76)
Synth LLM (iter 2)	25.74 (+1.36)	33.30 (+0.27)	26.33 (+3.20)	24.50 (+1.89)	64.57 (+0.31)
Synth LLM (iter 3)	24.94 (–0.80)	34.59 (+1.30)	24.73 (–1.60)	24.10 (–0.40)	64.20 (–0.37)
Synth LLM (iter 4)	25.30 (+0.36)	33.96 (–0.63)	25.37 (+0.64)	24.48 (+0.38)	64.67 (+0.46)
Synth LLM (iter 5)	25.55 (+0.25)	36.87 (+2.90)	23.36 (–2.01)	24.00 (–0.48)	63.69 (–0.98)

probability that the participant selected the gold-standard summary as the *best* summary decreased to 46.25% (37/80 total judgments), with individual accuracies ranging from 40% to 50%.

These results suggest that the model trained on synthetic data generated from only 10 gold examples produced summaries that were often indistinguishable from—or even preferred over—gold-standard summaries, aligning with our quantitative findings.

5.4. Comparison to Related Work

We evaluated our SYNPO fine-tuning approach on the problem list summarization (ProbSum) task, comparing it with state-of-the-art methods (Table 3). Our best model, fine-tuned on synthetic data with only 100 gold examples, achieved an RL-F1 score of 27.1, surpassing Li et al. (2023a)'s model, which scored 24.9 with 595 gold examples, and Park et al. (2024)'s 16-shot using GPT-4, which scored 25.1.

Our results are surpassed by Gao et al. (2024), who achieved a 30.0 RL-F1 score using supervised fine-tuning with knowledge graphs (KG) and FlanT5. However, in addition to differences in methodology, their use of both *Subjective* and *Assessment* information in prompts, compared to our use of only *Assessment*, may contribute to the performance gap. This limitation is discussed further in Section 6.1.

6. Discussion

Our findings highlight the effectiveness of SYNPO in improving model performance through iterative augmentations in synthetic data generation, even with limited gold-labeled data. Models fine-tuned with SYNPO-generated synthetic data consistently outperformed those trained on the full gold dataset, with performance peaking at around 100 gold examples. This suggests that SYNPO strikes a balance between data quantity and model generalization, an important advantage in contexts where obtaining large volumes of gold-standard data is challenging or expensive, particularly in medical research. An additional implication of our method is its potential to train LLMs to replicate the style of a specific medical professional using a limited set of their own labeled data to fine-tune a model on their local system, which could reduce the cognitive load on clinicians and streamline healthcare operations, all while ensuring sensitive patient data never leaves their respective machines. However, while SYNPO is promising for the ProbSum task, further research is needed to evaluate its broader applicability in diverse clinical and non-clinical NLP contexts.

Furthermore, variations in qualitative evaluation results from Section 4.2 emphasize the inherently subjective nature of assessing summary quality. Incorporating multiple evaluators and diverse metrics in the evaluation process to control for potential biases and maintain diversity is essential to capturing a broader range of perspectives and accurately assessing the *quality* of nuanced narratives.

Table 2: Performance of selected models trained on synthetic data generated using only 10 real examples after 5 SYNPO iterations, evaluated using ROUGE-L (RL), METEOR Score, and BERTScore. Numbers in parentheses are the increase/decrease compared to the previous row. Prior baselines from the literature are in Table 3.

Model	RL-F1	RL-Precision	RL-Recall	METEOR	BERTScore-F1	Average Char L
Base LLM	19.72	18.74	30.90	25.35	61.41	190
Gold LLM	25.25 (+5.53)	34.58 (+15.83)	31.21 (+0.31)	26.20 (+0.84)	63.91 (+2.50)	118 (–72)
Synth LLM (iter 0)	23.18 (–2.06)	28.79 (–5.78)	25.25 (–5.97)	24.08 (–2.12)	63.67 (–0.24)	81 (–37)
Synth LLM (iter 1)	25.20 (+2.02)	37.33 (+8.54)	24.37 (–0.88)	22.16 (–1.92)	63.94 (+0.27)	73 (–8)
Synth LLM (iter 2)	24.67 (–0.53)	36.78 (–0.55)	27.53 (+3.16)	23.64 (+1.48)	63.89 (–0.05)	84 (+11)
Synth LLM (iter 3)	26.72 (+2.05)	38.40 (+1.61)	29.68 (+2.15)	24.86 (+1.22)	64.79 (+0.91)	75 (–9)
Synth LLM (iter 4)	26.70 (–0.02)	38.29 (–0.10)	25.41 (–4.28)	23.44 (–1.42)	65.71 (+0.92)	62 (–13)
Synth LLM (iter 5)	25.38 (–1.32)	37.77 (–0.52)	26.93 (+1.52)	24.61 (+1.17)	63.91 (–1.79)	70 (+8)
Synth LLM (iter 6)	26.06 (+0.68)	36.54 (–1.23)	24.93 (–2)	24.35 (–0.25)	65.14 (+1.22)	57 (–13)
Synth LLM (iter 7)	26.19 (+0.13)	35.54 (–1)	24.64 (–0.30)	24.23 (–0.13)	65.61 (+0.47)	53 (–4)
Synth LLM (iter 8)	23.12 (–3.06)	38.31 (+2.77)	20.20 (–4.43)	19.26 (–4.97)	63.16 (–2.45)	41 (–12)
Synth LLM (iter 9)	24.93 (+1.81)	36.88 (–1.42)	22.18 (+1.97)	22.01 (+2.75)	64.27 (+1.11)	46 (+5)
Synth LLM (iter 10)	26.05 (+1.12)	38.10 (+1.21)	23.91 (+1.73)	23.66 (+1.65)	64.68 (+0.41)	50 (+4)
Synth LLM (iter 11)	27.11 (+1.06)	37.16 (–0.94)	25.32 (+1.41)	24.81 (+1.15)	65.60 (+0.92)	54 (+4)
Synth LLM (iter 12)	24.82 (–2.30)	35.43 (–1.73)	22.36 (–2.96)	22.44 (–2.37)	64.24 (–1.36)	51 (–3)
Synth LLM (iter 13)	26.27 (+1.45)	38.31 (+2.88)	23.40 (+1.04)	22.95 (+0.51)	65.06 (+0.83)	46 (–5)
Synth LLM (iter 14)	26.05 (–0.22)	36.65 (–1.66)	23.88 (+0.48)	23.34 (+0.39)	64.72 (–0.34)	51 (+5)
Synth LLM (iter 15)	25.14 (–0.90)	33.07 (–3.58)	25.79 (+1.91)	23.53 (+0.19)	63.26 (–1.46)	58 (+7)
Synth LLM (iter 16)	24.97 (–0.18)	33.20 (+0.13)	26.85 (+1.06)	22.80 (–0.73)	64.14 (+0.88)	68 (+10)
Synth LLM (iter 17)	25.28 (+0.32)	36.26 (+3.06)	23.03 (–3.82)	21.99 (–0.80)	64.71 (+0.57)	49 (–19)
Synth LLM (iter 18)	25.42 (+0.14)	34.05 (–2.20)	25.93 (+2.89)	23.72 (+1.73)	64.17 (–0.54)	67 (+18)
Synth LLM (iter 19)	25.99 (+0.57)	36.59 (+2.54)	24.05 (–1.88)	24.15 (+0.43)	65.25 (+1.08)	52 (–15)

6.1. Limitations

First of all, our reliance on ORPO as the main preference optimization algorithm being implemented for iterative augmentation may limit the model's exposure to diverse preferences and strategies to optimize such preferences; future work could explore other methods and potentially incorporate human feedback for active learning. The precision-recall trade-off observed in Section 5.1 highlights the need for future work to balance information accuracy and completeness, such as implementing modified reward functions or hybrid synthetic-real data approaches. Additionally, using a single LLM (Mistral) for both data generation and fine-tuning may introduce biases. As such, exploring multiple models or ensembles could improve data quality and diversity. Furthermore, our method's reliance on the *Assessment* component of medical notes alone is a limitation compared to gold-standard summaries created with access to the complete SOAP notes available in the dataset. This gap was evident in our qualitative evaluations, where participants struggled to determine if references to unmentioned conditions were due to model hallucinations or human access to additional SOAP components not provided to the model. For future work, we aim to incorporate the full SOAP note into our data inputs as more open-source models with extended context lengths become available (Liu et al., 2024).

Given that our method still requires gold-standard data, another concern is the potential for protected health information (PHI) leakage. This necessitates rigid privacy safeguards, especially for closed-source models on hardware that may not adhere to these standards. Additionally, due to its iterative nature, SYNPO requires more computational resources for fine-tuning and repeated iterations, which could limit its scalability in resource-constrained environments.

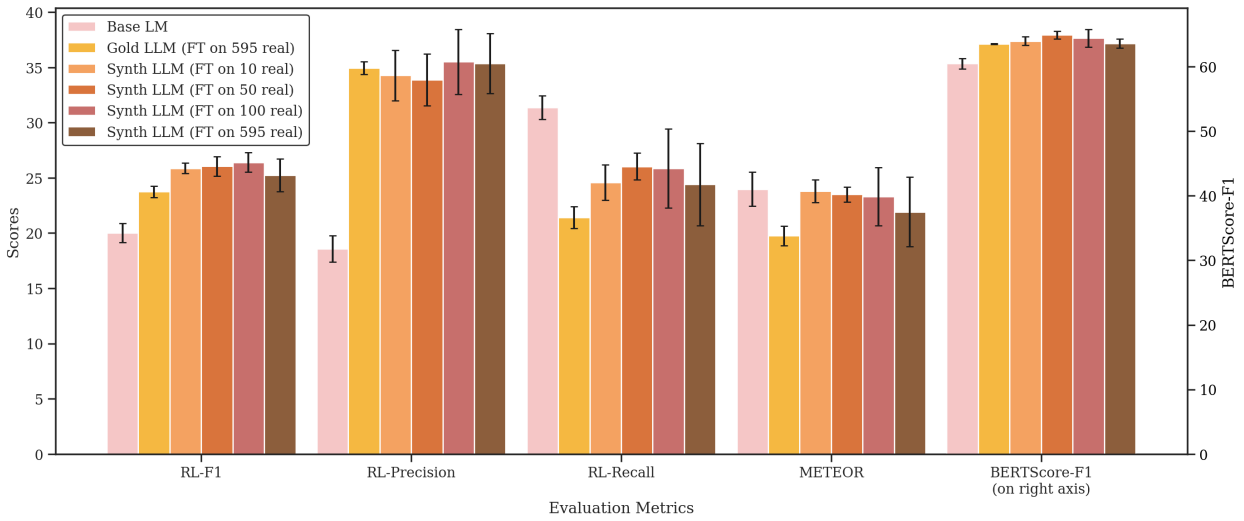

Figure 2: Impact of gold label mixtures on synthetic data performance. Average of best models for k-shot inference across 3 experiments. Base LLM: base language model; Gold LLM: Base model fine-tuned on the full gold dataset; Synth LLM (n gold): Base model fine-tuned on synthetic data generated using n gold examples.

Future research should explore ways to reduce the dependence on gold data and improve the computational efficiency of the iterative process.

7. Conclusion

In this paper, we introduce SYNPO, a novel approach that iteratively refines the generation of high-quality synthetic annotated data using LLMs. Our method enables LLMs fine-tuned on synthetic data generated with as few as 10 clinician expert-annotated examples to outperform models trained on a full dataset of 595 examples in clinical text summarization—demonstrating SYNPO's effectiveness in reducing the need for extensive manual annotation and accelerating model development. While our findings are promising within the ProbSum task, further research is needed to assess SYNPO's generalizability across different tasks and datasets. As we continue refining this approach, SYNPO holds the potential for advancing NLP models in the medical domain, ensuring high performance even in data-scarce environments.

8. Citations and Bibliography

Acknowledgments

This work was supported by the ... This Will be added for the camera ready version upon acceptance

References

Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, Bryan Catanzaro, Sharon Clay, Jonathan Cohen, et al. Nemotron-4 340b technical report. *arXiv preprint arXiv:2406.11704*, 2024.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

Wendy W Chapman, Prakash M Nadkarni, Lynette Hirschman, Leonard W D'Avolio, Guergana K Savova, and Ozlem Uzuner. Overcoming barriers to NLP for clinical text: the role of shared tasks and

Table 3: Comparison of RL-F1 (ROUGE-L F1) scores for the ProbSum task across different models and methods, including our SYNPO approach

Method	Gold Examples	Method	Model	RL-F1
Gao et al. (2024)	0	Zero-shot	GPT 3.5 Turbo	19.8
Park et al. (2024)	16	16-shots ICL	GPT-4	25.1
Li et al. (2023a)	595	SFT	FlanT5	24.9
Gao et al. (2024)	595	SFT+KG	GPT+FlanT5	30.0
Ours	10	SYNPO	Mistral 7B	26.4
Ours	50	SYNPO	Mistral 7B	26.6
Ours	100	SYNPO	Mistral 7B	27.1

the need for additional creative solutions. *Journal of the American Medical Informatics Association*, 18(5):540–543, 09 2011. ISSN 1067-5027. doi: 10.1136/amiajnl-2011-000465. URL https://doi.org/10.1136/amiajnl-2011-000465.

Shan Chen, Jack Gallifant, Marco Guevara, Yanjun Gao, Majid Afshar, Timothy Miller, Dmitriy Dligach, and Danielle S. Bitterman. Improving clinical nlp performance through language model-generated synthetic clinical data. 2024a. URL https://arxiv.org/abs/2403.19511.

Shan Chen, Marco Guevara, Shalini Moningi, Frank Hoebers, Hesham Elhalawani, Benjamin H. Kann, Fallon E. Chipidza, Jonathan Leeman, Hugo J W L Aerts, Timothy Miller, Guergana K. Savova, Jack Gallifant, Leo A. Celi, Raymond H. Mak, Maryam Lustberg, Majid Afshar, and Danielle S. Bitterman. The effect of using a large language model to respond to patient messages. *The Lancet Digital Health*, 6(6):e379–e381, 2024b.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models, 2024c. URL https://arxiv.org/abs/2401.01335.

Lizhou Fan, Lingyao Li, Zihui Ma, Sanggyu Lee, Huizi Yu, and Libby Hemphill. A bibliometric review of large language models research from 2017 to 2023, may 2024. ISSN 2157-6904. URL https://doi.org/10.1145/3664930. Just Accepted.

Maayan Frid-Adar, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Synthetic data augmentation using gan for improved liver lesion classification. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 289–293, 2018. doi: 10.1109/ISBI.2018.8363576.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, John Caskey, Brihat Sharma, Matthew M. Churpek, and Majid Afshar. Dr.bench: Diagnostic reasoning benchmark for clinical natural language processing. *Journal of Biomedical Informatics*, 138:104286, 2023. ISSN 1532-0464. doi: https://doi.org/10.1016/j.jbi.2023.104286. URL https://www.sciencedirect.com/science/article/pii/S1532046423000072.

Yanjun Gao, Ruizhe Li, Emma Croxford, John R Caskey, Brian W Patterson, Matthew M. Churpek, Timothy Miller, Dmitriy Dligach, and Majid Afshar. *Leveraging a medical knowledge graph into large language models for diagnosis prediction (preprint)*, Mar 2024. doi: 10.2196/preprints.58670.

Marco Guevara, Shan Chen, Spencer Angus Thomas, Tafadzwa L Chaunzwa, Idalid Franco, Benjamin H. Kann, Shalini Moningi, Jack M Qian, Madeleine H Goldstein, Susan M. Harper, Hugo J.W.L. Aerts, Guergana K. Savova, Raymond H. Mak, and Danielle S. Bitterman. Large language models to identify social determinants of health in electronic health records. *NPJ Digital Medicine*, 7, 2023. URL https://api.semanticscholar.org/CorpusID:260887020.

Daniel Han, Y X, and Y X. unslothai/unsloth: Finetune llama 3.1, mistral, phi gemma llms 2-5x faster with 80 URL https://github.com/unslothai/unsloth.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model, 2024. URL https://arxiv.org/abs/2403.07691.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3 (1), May 2016. doi: 10.1038/sdata.2016.35.

Dhiraj Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu Huang, Hector Yuen, et al. A study of bfloat16 for deep learning training. *arXiv preprint arXiv:1905.12322*, 2019.

Nicholas Lee, Thanakul Wattanawong, Sehoon Kim, Karttikeya Mangalam, Sheng Shen, Gopala Anumanchipali, Michael W Mahoney, Kurt Keutzer, and Amir Gholami. Llm2llm: Boosting llms with novel iterative data enhancement. *arXiv*, 2024.

Hao Li, Yuping Wu, Viktor Schlegel, Riza Batista-Navarro, Thanh-Tung Nguyen, Abhinav Ramesh Kashyap, Xiao-Jun Zeng, Daniel Beck, Stefan Winkler, and Goran Nenadic. Team:PULSAR at ProbSum 2023:PULSAR: Pre-training with extracted healthcare terms for summarising patients' problems and data augmentation with black-box large language models. In Dina Demner-fushman, Sophia Ananiadou, and Kevin Cohen, editors, *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 503–509, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.bionlp-1.49. URL https://aclanthology.org/2023.bionlp-1.49.

Rumeng Li, Xun Wang, and Hong Yu. Two directions for clinical data generation with large language models: Data-to-label and label-to-data. *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, 2023:7129–7143, 2023b. URL https://api.semanticscholar.org/CorpusID:266176903.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. Synthetic data generation with large language models for text classification: Potential and limitations. *arXiv preprint arXiv:2310.07849*, 2023c.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5, 2017.

Oren Melamud and Chaitanya P. Shivade. Towards automatic generation of shareable synthetic clinical notes using neural language models. *ArXiv*, abs/1905.07002, 2019. URL https://api.semanticscholar.org/CorpusID:158046910.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization. *arXiv preprint arXiv:2404.19733*, 2024.

Chaelim Park, Hayoung Lee, and Ok-ran Jeong. Leveraging medical knowledge graphs and large language models for enhanced mental disorder information extraction. *Future Internet*, 16(8), 2024. ISSN 1999-5903. doi: 10.3390/fi16080260. URL https://www.mdpi.com/1999-5903/16/8/260.

Raul Puri, Ryan Spring, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Training question answering models from synthetic data. *arXiv preprint arXiv:2002.09599*, 2020.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. Does synthetic data generation of llms help clinical text mining? *ArXiv*, abs/2303.04360, 2023. URL https://api.semanticscholar.org/CorpusID:257405132.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Jason A. Walonoski, Mark Kramer, Joseph Nichols, Andre Quina, Chris Moesel, Dylan Hall, Carlton Duffett, Kudakwashe Dube, Thomas Gallagher, and Scott McLachlan. Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *Journal of the American Medical Informatics Association : JAMIA*, 25:230 – 238, 2017. URL https://api.semanticscholar.org/CorpusID:3815968.

Benyou Wang, Qianqian Xie, Jiahuan Pei, Zhihong Chen, Prayag Tiwari, Zhao Li, and Jie Fu. Pre-trained language models in biomedical domain: A systematic survey. *ACM Comput. Surv.*, 56(3), oct 2023. ISSN 0360-0300. doi: 10.1145/3611651. URL https://doi.org/10.1145/3611651.

Huizi Yu, Lizhou Fan, Lingyao Li, Jiayan Zhou, Zihui Ma, Lu Xian, Wenyue Hua, Sijia He, Mingyu Jin, Yongfeng Zhang, Ashvin Gandhi, and Xin Ma. Large language models in biomedical and health informatics: A bibliometric review, 2024.

Appendix A. Experiment Details

A.1. Inference Prompt

```
Given an assessment, please summarize
the patient condition and output the
summary.
###
Here are some examples:
Assessment: {}
Summary: {}
Assessment: {}
Summary: {}
...
###
Given the following assessment, please
summarize the patient condition and
output the summary:
Assessment: {}
Summary:
```

A.2. Alpaca-style Prompt

Adapted from Taori et al. (2023)

```
Below is an instruction that describes
a task, paired with an input that
provides further context. Write a
response that appropriately completes
the request.
### Instruction:
{}
### Input:
{}
### Response:
{}
```

A.3. Fine-Tuning Prompt

Adapted from A.2

```
Below is an instruction that describes
a task, paired with an input that
provides further context. Write a
response that appropriately completes
the request.
### Instruction:
Given an assessment from doctor notes,
please summarize the patient condition
and output the summary.
### Input:
Assessment: {}
### Response:
Summary:
```

A.4. Synthetic Data Generation Prompt

```
Generate {num_to_gen} unique sets of
doctor's assessments, subjective notes,
objective notes, and summaries of the
assessments.
###
Here are some examples:
Assessment: {}
Summary: {}
Assessment: {}
Summary: {}
...
###
Generate {num_to_gen} new, unique pairs.
Do not repeat examples or previously
generated content. Do not include any
additional text or numbering. Follow
this format strictly, with each pair
separated by a new line:
Assessment: [Doctor's assessment of
the patient]
Summary: [Concise summary of the
assessment]
```

A.5. SYNPO Fine-Tuning Prompt To Improve Synthetic Data Generation

Adapted from A.2

```
Below is an instruction that describes
a task, paired with an input that
provides further context. Write a
response that appropriately completes
the request.
### Instruction:
Generate a pair of medical assessments
from doctors' notes and corresponding
summaries of these assessments.
### Input:
The Assessment should briefly describe
the patient's age, gender, relevant
medical history, current symptoms, and
any significant clinical findings.
Ensure that the Summary lists key
diagnoses or concerns and accurately
reflects the Assessment.
Follow this format strictly, and do not
include any additional information:
Assessment: [Doctor's assessment of
the patient]
Summary: [Concise summary of the
assessment]
```

A.6. Training Setup

LoRA is particularly advantageous in our setup because it updates only a small fraction (1-10%) of the model's parameters, significantly reducing memory requirements while maintaining model effectiveness. In our implementation, we set the LoRA rank to 8, which determines the size of the low-rank matrices used for adaptation, with higher ranks potentially capturing more complex relationships at the cost of increased computational requirements. This value was chosen to balance efficiency and performance, ensuring that enough model capacity is retained for learning meaningful representations without incurring excessive computational costs.

The LoRA configuration targeted specific modules in the transformer architecture—namely, `q_proj`, `k_proj`, `v_proj` (Query, Key, and Value projection layers in the attention mechanism), `o_proj` (Output projection), `gate_proj` (Gating Mechanism Projection), `up_proj`, and `down_proj` (Feedforward Network Projections). These modules are critical in managing the self-attention mechanism and the feedforward

network in our transformer-based LLM architecture to optimize the model's ability to handle the complexity of clinical text while keeping the overall adaptation lightweight. Additional LoRA parameters included a `lora_alpha` of 16, which scales the LoRA updates, and a `lora_dropout` of 0, optimized for performance.

For the training process, we employed the AdamW optimizer (Loshchilov et al., 2017) with a learning rate of $2e-5$, weight decay of 0.2, and a maximum gradient normalization of 0.3. We used a cosine learning rate scheduler with 5 warmup steps. Training was conducted for a maximum of 3 epochs, with early stopping implemented based on the evaluation loss to prevent overfitting to the training set, with a patience of 3 steps. The batch size was set to 8 for both training and evaluation. To ensure reproducibility, we set a random seed of 3407 for all randomized processes.

Training of our models was distributed on one NVIDIA RTX 4080 machine or two RTX 3090 Ti GPUs machine with CUDA 12.0. As such, we were able to leverage BF16 (Brain floating point 16) precision to maintain a wide dynamic range equivalent to FP32 (32-bit floating point), ensuring stable convergence across iterations and eliminating the need for hyper-parameter adjustments typically required when using traditional FP16 (16-bit floating point) formats, with a slight tradeoff in precision (Kalamkar et al., 2019).