Assessing the Viability of Open Large Language Models for Clinical Documentation: A Real-world Study in German Healthcare

Felix Heilmeyer, Daniel Böhringer, Thomas Reinhard, Sebastian Arens, Lisa Lyssenko, Christian Haverkamp

Submitted to: JMIR Medical Informatics
on: April 17, 2024

Disclaimer: © The authors. All rights reserved. This is a privileged document currently under peer-review/community review. Authors have provided JMIR Publications with an exclusive license to publish this preprint on it's website for review purposes only. While the final peer-reviewed paper may be licensed under a CC BY license on publication, at this stage authors and publisher expressively prohibit redistribution of this draft paper other than for review purposes.

Table of Contents

Assessing the Viability of Open Large Language Models for Clinical Documentation: A Real-world Study in German Healthcare

Felix Heilmeyer[1]; Daniel Böhringer[2] Dr med; Thomas Reinhard[2] Dr med; Sebastian Arens[2] Dr med; Lisa Lyssenko[1] Dr; Christian Haverkamp[1] Dr med

[1]Medical Center University of Freiburg Institute for Digitization in Medicine Freiburg im Breisgau DE
[2]Medical Center University of Freiburg Eye Center Freiburg im Breisgau DE

Corresponding Author:
Felix Heilmeyer
Medical Center University of Freiburg
Institute for Digitization in Medicine
Breisacher Str. 153
Freiburg im Breisgau
DE

Abstract

Background: The use of Large Language Models (LLMs) as writing assistance for medical professionals is a promising approach to reduce the time required for documentation, but there may be practical, ethical, and legal challenges in many jurisdictions complicating the use of the most powerful commercial LLM solutions.

Objective: In this study, we assess the feasibility of using non-proprietary LLMs of the Generative Pretrained Transformer (GPT) variety as writing assistance for medical professionals in an on-premise setting with restricted compute resources, generating German medical text.

Methods: We train four 7B parameter model variants for our task and evaluate their performance using a powerful commercial LLM, namely Anthropic's Claude-v2 as a rater. Based on this, we select the best performing model and evaluate its practical usability with two independent human raters on real world data.

Results: In the automated evaluation with Claude-v2 BLOOM-CLP-German, a model trained from scratch on German text, achieved the best results. In the manual evaluation by human experts, 95 of the 102 reports generated by that model were evaluated as usable as is or with only minor changes by both human raters (93.1%).

Conclusions: The results show that even with restricted compute resources it is possible to generate medical texts that are suitable for documentation in routine clinical practice, but that language issues need to be considered when processing non-English text.

(JMIR Preprints 17/04/2024:59617)
DOI: https://doi.org/10.2196/preprints.59617

Preprint Settings

Original Manuscript

Assessing the Viability of Open Large Language Models for Clinical Documentation: A Real-world Study in German Healthcare

Felix A. Heilmeyer[1*], Daniel Böhringer[2*], Thomas Reinhard[2], Sebastian Arens[2], Lisa Lyssenko[1], Christian Haverkamp[1]

[1]Institute for Digitization in Medicine, Medical Center University of Freiburg, Breisacher Str. 153, Freiburg, D-79110, Germany

[2]Eye Center, Medical Center University of Freiburg, Killianstraße 5, Freiburg, D-79106, Germany

*Corresponding authors: felix.heilmeyer@uniklinik-freiburg.de;

daniel.boehringer@uniklinik-freiburg.de

Abstract

Background: The use of Large Language Models (LLMs) as writing assistance for medical professionals is a promising approach to reduce the time required for documentation, but there may be practical, ethical, and legal challenges in many jurisdictions complicating the use of the most powerful commercial LLM solutions.

Objective: In this study, we assess the feasibility of using non-proprietary LLMs of the Generative Pretrained Transformer (GPT) variety as writing assistance for medical professionals in an on-premise setting with restricted compute resources, generating German medical text.

Methods: We train four 7B parameter four models with three different architectures for our task and evaluate their performance using a powerful commercial LLM, namely Anthropic's Claude-v2 as a rater. Based on this, we select the best performing model and evaluate its practical usability with two independent human raters on real world data.

Results: In the automated evaluation with Claude-v2 BLOOM-CLP-German, a model trained from scratch on German text, achieved the best results. In the manual evaluation by human experts, 95 of the 102 reports generated by that model were evaluated as usable as is or with only minor changes by both human raters (93.1%).

Conclusions: The results show that even with restricted compute resources it is possible to generate medical texts that are suitable for documentation in routine clinical practice. However, the target language should be considered in the model selection when processing non-English text.

Keywords: Large Language Models; medical documentation; writing assistance for physicians

1) Introduction

1.1 Background

Physicians are often overloaded with documentation requirements, including writing a doctor’s note, a summary of a patient’s visit. An analysis of clinical software log files showed that interaction with Electronic Health Records (EHR) constitutes a large portion of physicians’ daily work, approximately a fourth of which is spent writing documentation [1]. Completion of the documentation in the EHR is perceived as a tedious task, which is often done after work hours [1]. More time spent on documentation in after work hours has been shown to be associated with burnout and decreased work-life satisfaction [2].

A promising approach to reduce the time required for documentation is the use of writing assistance based on Large Language Models (LLMs). In a feasibility study, the authors trained previous generation LLMs (GPT-2 and GPT-Neo) to complete text in medical records [3]. They concluded that the models could be used in medical charting, but still have some room for improvement. A large source of error were abrupt changes in the topic, which is common in documentation of EHRs.

With recent advances in LLM technology and the release of ChatGPT, LLMs have seen widespread adoption in assisting professionals produce text for communication or documentation purposes. For example, under the Copilot brand, Microsoft is building generative artificial intelligence (AI) capabilities into their widely used Office application suite to assist in business use cases. This leads us to believe that current generation LLMs could also provide valuable assistance in the healthcare sector.

1.2 Challenges in the Use of LLMs in the Healthcare Sector

Among the best performing LLMs according to the continuously updated Holistic Evaluation of Language Models (HELM) [4] at Stanford University are currently commercial offerings from companies such as OpenAI or Anthropic. With these offerings, the models run on the providers’ infrastructure and are accessible via an application programming interface (API). However, these services cannot be used in a clinical context without further consideration.

First, in many countries, the services do not meet the legal requirements for processing protected health information (PHI). In some jurisdictions, legal and regulatory frameworks mandate that data originating from healthcare providers must be processed within the country’s borders or even on-premise. This is particularly problematic for European countries, as the European Union’s General Data Protection Regulation (GDPR) prohibits the transfer of PHI to datacenters in the US, where most providers are located.

Second, clinical software must be thoroughly validated before it is released to end users, and in some cases it is even subject to the Medical Device Regulation. This conflicts with the update policy of providers of commercial AI solutions. The scope of model updates is usually communicated only a few weeks in advance, for example, two weeks in the case of OpenAI [5]. This would not be a problem if these updates were only additive in functionality, but the opaque nature of current LLMs also means that improvements to some aspects of model performance might unexpectedly negatively affect the performance on other tasks [6]. The use of fixed model versions, as offered by some providers, is not practicable in the long term, as older models are often removed after the release of updates; in the case of OpenAI after three months [5].

1.3 Training Non-Proprietary AI Models for Medical Text

An alternative is the use of non-proprietary AI models. In these models, the architecture as well as the trained parameters are available to the user. This solves the aforementioned problems by giving the user the option to train and deploy these models on any infrastructure and fully control any changes to it.

One of the largest pre-trained LLMs is Generative Pretrained Transformer (GPT) models that enable model training with limited datasets. There are several approaches to applying GPT models to a task. One common approach is to use a very large model that is trained primarily with general text corpora and include instructions for the task in the input for the model, the so-called prompt. This is sometimes called incontext learning (ICT) or, depending on whether examples are provided, zero-shot or few-shot learning.

ICT works reasonably well on tasks that have a good representation in the base models’ training corpus. However, the structure and content of clinical notes differ significantly from the general purpose text corpora used to train most publicly available LLMs. Even including biomedical text from publications, such as PubMed articles, in the training data could only have minor effects on model performance compared to training on clinical text [7–9]. In [9] the authors compare ICT and multiple alternatives such as: (a) training from scratch on a clinical corpus, (b) continuing training a pre-trained model on clinical text and then fine-tuning for the downstream task, or (c) directly training the GPT for the downstream task without further pre-training. They show that relatively small specialized clinical models substantially outperform all in-context learning approaches and conclude that pretraining on clinical text allows for smaller, more parameter-efficient models.

One fact that must be taken into account when using GPT models in a clinical context is that the pre-trained models have now become very large. Complete finetuning, in which all model parameters are re-trained on the task-specific data, is therefore becoming less and less feasible. This is particularly the case if the models have to be trained on site for legal and/or economic reasons. The computing power available here is usually limited, which restricts the size of the models that can be trained. Accordingly, the choice of models is a trade-off between training time and costs, model accuracy, and maximum sequence length.

One possibility to address the problem of limited working memory is the Low Rank Adaptation (LoRA) technique [10]. Here, all the model weights are frozen and only a few very small additional low rank matrices are added to the query and key parameter matrices of the transformer attention heads and subsequently optimized. This reduces the number of trainable parameters by 10,000 times and the GPU memory requirement by 3 times. Recently, training of quantized models became possible by combining LoRA with quantization [11]. With QLoRA a frozen quantized model is finetuned by optimizing added low-rank adapters at 16 bit floating-point precision. The QLoRA technique also introduced additional memory saving mechanism such as the 4 bit Normal Float (NF4) data type for quantization and paged optimizers [11].

1.4 Aim of the Study

In this study we assess the feasibility of using non-proprietary LLMs of the Generative Pretrained Transformer (GPT) variety as writing assistance for medical professionals in an on-premise setting with restricted compute resources, generating non-English medical text. We train four models with three different architectures for our task using the HuggingFace Transformers framework [12] and explore their performance using a powerful commercial LLM, namely Anthropic's Claude-v2 as a rater. Based on this, we select the best performing model and evaluate its practical usability with two independant human raters on real world data.

2) Methods

The study was implemented in the outpatient clinic of the Eye Center at Medical Center, University of Freiburg, Germany and approved by the responsible ethical review committee (registration number: 23-1444-S1).

The target for assistive text generation was the final part of the medical documentation of an examination or treatment, the so-called epicrisis report. In this report, the doctors write a structured compilation of the information so far documented in the EHR in text form. It contains the relevant medical information of the case and usually consists of three sections: (1) main diagnosis or the patient's reason for visit, (2) therapeutic procedures and/or medication, and (3) recommendations for further intervention and/or need for a follow-up appointment.

2.1 Data

2.1.1 Data Source and Description

The data pool used for training the models were the EHR records of 82.482 unique patient encounters that span approximately 10 years of clinical practice. The EHR record of an encounter contains all digital information about a patient's examination or treatment in the outpatient clinic, which offers specialist, emergency and follow-up care. The data is collected in various ways over the patient's visit. Support staff record basic information in structured forms, doctors document the medical history, symptoms and previous or planned treatments in text notes, and diagnostic data from electronic devices is mainly stored in numeric format. The final epicrisis report consists of a stand-alone text, which is filed alongside all other information in the EHR record.

The whole training dataset amounts to approximately 140 MB of uncompressed text in UTF-8 encoding or approx. 29M to 33M tokens, depending on the tokenizer model used. A data set of 509 patient encounters that occurred after the training set date cutoff was set aside for comparison of model performance in the evaluation. The complete data set consists of German text. The examples used in this paper were translated into English by the authors of the paper.

2.1.2 Preprocessing and Formatting

For the LLM training all available data in the EHR record were concatenated into one continuous text sequence per encounter. The types of information were separated by newlines and prefixed with a descriptor such as ’History’ or ’Pressure Measurement’ to form the prompt. If no data was documented in a section, it was left empty. The order of sections matched the order in which the fields are displayed to users in the EHR software interface. The last section of each text sequence was the epicrisis report. If there were separate records for each eye, the individual records were additionally prefixed with an abbreviation indicating the side.

Task training was implemented by inserting special tokens to mark the text to be generated by the final models, i.e. the epicrisis report. Each text sequence starts with a special token indicating the beginning of the input data recorded during the patient visit, i.e. all other information in the EHR record. A second special token is inserted before the epicrisis report, indicating the start of the generation task. In the training data, this token is followed by the actual report of the attending physician. The text sequence ends with a "Stop of Sequence" token, which indicates that the model should end the generation process.

For instruction tuned models the text sequence was prefixed with a so-called system message enclosed in special tokens indicating instructions for the model, reading as follows: ’You are an experienced doctor in a German eye hospital. Your writing style is concise, accurate, and respectful. You are writing a short note in German to a colleague about a patient. The letter should contain the provided information.’

2.2 Models

In the selection of models from openly available pretrained models, we considered hardware cost, feasibility of the training process, language aspects, and performance benchmark results, such as Stanford’s HELM [4] and the Open LLM Leaderboard on HuggingFace [13]. Most LLMs are predominantly trained in English texts, and currently there is no model that contains a greater amount of medical text. Consequently, we chose the following three models:

LLaMA At the start of this study, Meta AI’s LLaMA model was among the top performers on several open LLM benchmarks. In contrast to some of its competitors, its training corpus also contains some German text, but no clinical content [14]. Since then, more powerful models have been released, but LLaMA still achieves competitive results on many benchmarks.

LLaMA-2-Chat During our experiments Meta AI released the successor to LLaMA [15]. Together with the updated base model, they also released an instruction-tuned model aligned with human preferences using reinforcement learning, similar to how ChatGPT was based on GPT-3 [16]. We chose this model to investigate the potential advantage of using an instruction-tuned model.

BLOOM-CLP-German This model is designed for tasks in German, based on the BLOOM architecture from [17]. It was initialized with the novel Cross Lingual and Progressive Transfer Learning (CLP) technique [18], which uses information from a small model trained in a target language and a larger model in a source language. This considerably reduces the training needed to achieve performance on par with that of a model trained from scratch. Although the model is still severely undertrained for its size[19], we included it to study the potential performance gains achieved by a model with a training corpus closer to the target text material.

2.3 Training

We restrict our training setup to 8x NVIDIA RTX 3090 24 GiB consumer grade GPUs in a single host. We load and train our models using the ’transformers’ Python library by HuggingFace [12] with the PyTorch [20] backend. Data are preprocessed employing HuggingFace’s ’datasets’ Python library [21]. Distributed training on multiple GPUs is implemented via the ’accelerate’ Python library [22].

For each training process, we randomly sample 5% of the training data as validation data. We regularly evaluate training loss on the validation set during training, about 20 times per epoch. We stop training when the validation loss does not improve in 10 evaluation steps. This amounts to around 13 epochs for most models.

2.3.1 Memory Optimization

For fine-tuning the model for our task we employ the LoRA at full 16 bit precision and QLoRA[11] at reduced Normal Float 4bit (NF4) precision techniques. Reducing the precision also reduces the memory usage and allows for longer input text sequences with the available memory. With this, we explore the trade-off between computational precision and input context size.

Additionally, we use two methods to trade reduced memory requirements for computation time. First, we use gradient checkpointing, a technique that recomputes some network activations during the backward pass on the fly instead of caching them in memory. Second, we use the Zero Redundancy Optimizer (ZeRO) technique [23], which includes memory savings achieved by reducing redundancy when training on multiple GPUs, as well as offloading some tasks to the CPU, both at the cost of communication overhead. Both make the training process considerably slower, but should not impact task performance of the resulting model.

Specifically, we trained the following model variants:

- LLaMA with LoRA at FP16 precision
- LLaMA 2 Chat with QLoRA at NF4 precision
- BLOOM-CLP German with QLoRA at NF4 precision
- BLOOM-CLP German with LoRA at FP16 precision

2.4 Inference

At their core, the decoder part of the Transformer architecture models a probability distribution for the next token given a sequence of input tokens. Both, the composition of the initial input tokens and the method of choosing the next token from the produced probability distribution can have a big impact on the quality of the final result.

2.4.1 Completion Prefixing

At inference time, the model receives an input text sequence, often called the prompt. It consists of the input data, as described in section 2.1.2, followed by a special token indicating that the subsequent text should be an epicrisis report. In other words, the model receives a text sequence containing all information from an EHR record except for the attending physician's epicrisis report and is asked to write this report, i.e. to generate a text that corresponds in content, structure and form to the epicrisis reports included in the training data. However, in the qualitative analysis of our initial findings, we found that in some cases the models attempted to continue with the recorded data rather than start writing a final report.

In an effort to improve results without retraining our models, we introduce a simple form of prompt tuning by adding a static suffix to the prompt, i.e. forcing the model to begin the generated text with the words ’During today’s visit...’. This suffix represents the typical beginning of the epicrisis report, as almost all reports written by doctors in the training dataset start with some variation of these words. We hope that this gives the models an additional signal to complete the text with a summary and recommendations instead of trying to invent more ’facts’ about the patient’s stay. We report and compare the evaluation results on reports generated with and without the completion prefix.

2.4.2 Contrastive Search

For a given input sequence, the trained transformer model produces a probability distribution for the next token. Simply choosing the token with the highest probability often produces text that lacks coherence and diversity. Techniques that maximize the probability over multiple tokens (e.g., beam search) or stochastic sampling can enhance coherence and diversity but are not targeted at the problem of repetition that is common to the type of highly standardized text generated in this study. We therefore employ a more recently introduced technique, called Contrastive Search, which has been shown to encourage diversity and produce coherent results while reducing repetitiveness [24, 25].

2.5 Evaluation

Evaluating the quality of generated natural language text using (preferably multiple) human raters is costly and time-consuming. Especially if the rating process requires specialized domain knowledge, like in this study. On the other hand, there is no obvious way to automate this process. An interesting idea is to use larger and more powerful language models to rate the quality of the output. This technique has recently been employed in some publications in the LLM space, for example in the creation of the LLaMA-2 model and in evaluating the performance of QLoRA training [11, 15]. Large commercial language models such as OpenAI’s GPT-4 and Anthropic’s Claude-v1 model have been shown to achieve agreement rates with human raters of up to 80% when evaluating the output of other models [26].

2.5.1 Automated Evaluation with Claude-v2

We evaluate the generated text in a two-step process using Claude-v2 by comparing the generated text to the epicrisis reports that were written by physicians for 509 individual patient encounters. In the first step, we extract the text passages that contain relevant information for each of the three main categories of information: (1) main diagnosis or patient's reason for visit, (2) therapeutic procedures and/or medication, and (3) recommendations for further intervention and/or need for a follow-up appointment. In a second step, for each case and category separately, we ask Claude to evaluate whether the extracted passage from the generated report matches the passage extracted from the report written by a human.

2.5.2 Human Evaluation

The suitability of the generated text by the best performing model is evaluated by two independent expert senior physicians. For this purpose, the raters are presented with the basic data from the documentation of 102 patients as well as both versions of the report: the one written by the attending physician and the computer-generated version. The raters assess whether the computer-generated version is suitable as a text template and could be used without major changes.

3) Results

3.1 Model Performance

Table 1 shows the percentage of reports in the test set in which the models matched the extracted diagnosis, follow-up, and therapy recommendation. The highest agreement rates with reports written by a doctor were achieved by the BLOOM-CLP-German model, followed by LLaMA-2 and LLaMA. The ranking was consistent across all the diagnosis, follow-up, and therapy dimensions. On average, the models achieved the highest scores in the diagnosis dimension followed by the therapy and follow-up dimensions.

Table 1 Fraction of reports in the test set where the models match the information extracted from the text written by a doctor.

Model Category	bloom-clpfp16	bloom clpfp16 prefix	bloom clpqlora	bloom clpqlora prefix	llama-2 qlora	llama2 qlora prefix	llama fp16	llama fp16 prefix	Mean
Diagnosis	50.10 %	44.01 %	55.40 %	45.78 %	34.38 %	32.81 %	31.24 %	16.50 %	38.78 %
Follow-Up	41.45 %	32.02 %	43.42 %	33.79 %	36.94 %	31.24 %	34.97 %	21.02 %	34.36 %
Therapy	43.81 %	37.33 %	50.69 %	42.44 %	36.54 %	36.35 %	29.67 %	13.95 %	36.35 %
Mean	45.12 %	37.79 %	49.84 %	40.67 %	35.95 %	33.46 %	31.96 %	17.16 %	

Of the BLOOM-CLP-German variants trained with full floating point 16-bit precision LoRA and reduced NF4 integer precision QLoRA the latter achieved slightly higher agreement rates (see Figure 1). In contrast to our intuition, prefixing the model prompt at inference time (see section 2.4.1) slightly reduced the performance across all models rather than improving it.

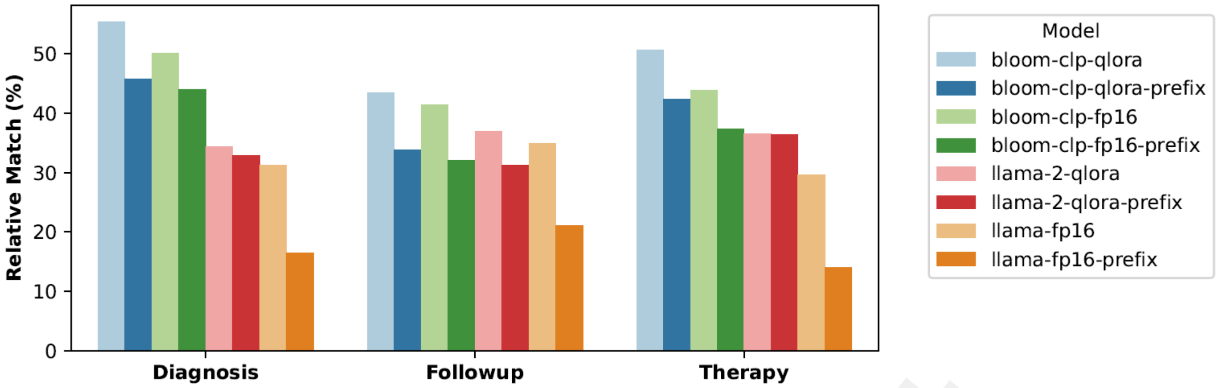


Figure 1 Fraction of reports in the test set where the models matched the extracted diagnosis, followup, and therapy recommendation of the doctor. All data are extracted using Claude-v2 (see section 2.5.1). For some models variants are trained using either full floating point 16-bit precision LoRA, indicated by the suffix ’fp16’ in the legend, or using NF4 integer precision QLoRA, indicated by the suffix ’qlora’. For all models we generate and evaluate two times: once using the standard prompt and once using a completion prefix (see section 2.4.1), indicated by the word ’prefix’ in the figure legend.

3.2 Human Evaluation

A total of 102 reports generated by the BLOOM-CLP German model trained with QLoRA at NF4 precision were rated for suitability by two independent expert senior physicians. 95 of the 102 reports were evaluated as suitable by both raters (93. 1%), which means that computer-generated reports could be used in this form or with minor changes. Only seven of the reports (6.9%) were rated as unsuitable by at least one of the raters. Cohen’s κ was run to determine the interrater reliability. There was moderate agreement between the two physicians’ judgments, $\kappa = .582$ (95% CI, .217 to .947), $p < .001$.

The 7 reports that were rated as unsuitable show different anomalies. In three of the reports, the model was caught in a loop of repeating nonsensical word sequences, for example, “we recommend local therapy with Bepanthen eye ointment 5x daily on both sides for 5–7 days, then 1x daily on both sides for 5–7 days, then 1x daily on both sides for 5–7 days, then 1x daily on both sides for 5-7 days, etc.” (26 repetitions). In one case, there is no text output because the patient’s appointment did not take place. Only in three reports are content-related aspects decisive. In one case, the main diagnosis is not mentioned, in one case information is missing in the treatment recommendation, and in one case the time given for the follow-up appointment is incorrect.

4) Discussion

Despite being severely undertrained compared to both LLaMA models, the BLOOMCLP-German model achieved the best performance in our experiments. This suggests that a better alignment of the base model with the reports’ language might be more important than a longer training time. We speculate that the German vocabulary in the model’s tokenizer better captured domain semantics compared to the multilingual tokenizers. Additionally, the model might have profited from a larger maximum input sequence length given the limited memory. This is an effect of the smaller token per character ratio of a tokenizer with a better alignment to the text's language.

Because its vocabulary is closer to our data, BLOOM-CLP-German's tokenizer encodes up to 30% fewer tokens for the same input text compared to LLaMA's tokenizer. This means that we can fit more information into the context window, training, and inference consume about half as much memory, and inference is about twice as fast. This makes for significant cost reductions compared to models with a multi-language tokenizer.

Of both BLOOM variants trained with LoRA and reduced QLoRA precision, the latter performed better in our analysis. This suggests that the reduced precision is more than offset by the bigger maximum input sequence length given the memory constraints. We surmise that capturing more context in the model input outweighs compute-optimal training or precision.

In contrast to our intuition, forcing the models to start the generated text with a predefined prefix did not improve the results. We speculated from our manual testing that this technique might eliminate some edge cases where the models sometimes start generating text completely unrelated to the input sequence. While this might still be true, the prefix also might have impacted the models' ability to flexibly react to the input. Therefore, reducing quality in more cases than improving it.

4.1 Feasibility of non-proprietary on-site AI

Our manual evaluation clearly shows that it is possible to provide helpful writing assistance using non-proprietary on-site AI technologies. Most of our test samples were rated useful as is or with only minor modifications. Additionally, qualitative analysis of samples rated as unusable showed that these were edge cases where the model produced no output or text that was easily identifiable as an anomaly. Only in very few reports were content-related aspects decisive, i.e. the model omitted major details or produced factually incorrect information.

While legal and ethical concerns, as discussed in section 1.2, currently may prevent many healthcare providers in European countries from using proprietary AI assistance for charting the solution presented in this study should be feasible for most of them. Non-proprietary models allow for flexible model deployment to comply with data protection requirements. Full control over the model also addresses legal concerns regarding software certification and some ethical concerns, because these models can be more easily inspected regarding potential biases.

In this study we chose model sizes around 7B parameters. In comparison, GPT3, the model that powered the first version of ChatGPT, has 175B parameters. With careful optimization of trade-offs between training time and cost, model precision, and maximum sequence length, we show that it is still possible to provide helpful writing assistance even with a much smaller model. At our chosen model scale operating, the models should be economically accessible to many healthcare providers or local service providers making it easier to comply with local regulation and reducing possible dependence on external or foreign service providers.

4.2 Limitations

Due to the limited availability of compute time, we were unable to test all combinations of model and training modalities. LLaMA-2 was only trained using QLoRA and LLaMA only using LoRA, limiting possible comparisons between the base models. Similarly, we only included the instruction-tuned variant of LLaMA-2 and cannot compare to the base model without instruction tuning.

The limited training of the BLOOM model probably affected its accuracy. On the flip side, this limitation highlights the importance of the language alignment with the under-trained BLOOM model outperforming both LLaMA models.

While our human raters evaluated our chosen model's outputs favorably, this happened in dedicated research settings. It remains to be shown whether AI writing assistance is still perceived as helpful in a real clinical setting or if the additional mental load caused by having to check the AI's output outweighs its usefulness.

5) Conclusions

In conclusion, this work demonstrates the feasibility of localized AI assistance for clinical note generation using small-scale non-proprietary models. Our results highlight the advantages of language-specific model tuning, providing a promising direction for future research. Especially when considering the significant speed and cost advantages of the language-specific model.

5.1 Future Work

Moving forward, leveraging German clinical corpora for pretraining could provide useful in-domain semantics. Techniques such as CLP fine-tuning can enable the utilization of such data with lower compute requirements. In a future study, we will explore the usage of our models in a real world setting.

6) Author Contributions

FH prepared training data, trained the models, performed the automated evaluation, analyzed evaluation results, and wrote the manuscript. DB initiated the project, collected and prepared training data, performed qualitative evaluation of results during model training, contributed to the quantitative human evaluation, and to writing the manuscript. TR supervised the project in the Eye Center. SA organized and contributed to the human evaluation. LL analyzed the results of the human evaluation and was a major contributor to the writing of the manuscript. CH supervised the project at the Institute for Digitization in Medicine. All authors read and approved the final manuscript.

7) Code Availability

The Code used to train and evaluate the models was archived on zenodo.org

8) Acknowledgements

The authors gratefully acknowledge the Platform for Infrastructure, Education and Research in AI (AI-PIER) at the University of Freiburg Medical Center for providing computing resources.

9) References

1. Robertson SL, Robinson MD & Reid A. Electronic Health Record Effects on Work-Life Balance and Burnout within the I3 Population Collaborative. Journal of Graduate Medical Education; 2017 (9): 479–484. doi:10.7326/M18-3684

2. Overhage JM & McCallie D. Physician Time Spent Using the Electronic Health Record During Outpatient Encounters: A Descriptive Study. Annals of Internal Medicine; 2020 (172): 169–174. doi:10.7326/M18-3684

3. Sirrianni J, Sezgin E, Claman D & Linwood SL. Medical Text Prediction and Suggestion Using Generative Pretrained Transformer Models with Dental Medical Notes. Methods of Information in Medicine; 2022 (61): 195–200. doi:10.1055/a-1900-7351

4. Liang P et al. Holistic Evaluation of Language Models. arXiv; 2022. doi:10.48550/arXiv.2211.09110

5. OpenAI Platform – Deprecations. https://platform.openai.com/docs/deprecations/; 2023.

6. Chen L, Zaharia M & Zou J. How is ChatGPT’s behavior changing over time? arXiv; 2023. doi:10.48550/arXiv.2307.09009

7. Moradi M, Blagec K, Haberl F & Samwald M. GPT-3 Models are Poor FewShot Learners in the Biomedical Domain. arXiv; 2022. doi:10.48550/arXiv.2109.02555

8. Yang X et al.. A large language model for electronic health records. npj Digital Medicine; 2023 (5): 194. doi: 10.1038/s41746-022-00742-2

9. Lehman E et al. Do We Still Need Clinical Language Models? arXiv; 2023. doi: 10.48550/arXiv.2302.08091

10. Hu EJ et al. LoRA: Low-Rank Adaptation of Large Language Models arXiv; 2021. doi:10.48550/arXiv.2106.09685

11. Dettmers T, Pagnoni A, Holtzman A & Zettlemoyer L. QLoRA: Efficient Finetuning of Quantized LLMs arXiv; 2023. doi: 10.48550/arXiv.2305.14314

12. Wolf T et al. Transformers: State-of-the-Art Natural Language Processing in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations; 2020: 38–45. doi:10.18653/v1/2020.emnlp-demos.6

13. Beeching, E et al. Open LLM Leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard; 2023.

14. Touvron H et al. LLaMA: Open and Efficient Foundation Language Models arXiv; 2023. doi:10.48550/arXiv.2302.13971

15. Touvron H, Martin L & Stone K. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv; 2023. doi:10.48550/arXiv.2307.09288

16. Christiano, P et al. Deep reinforcement learning from human preferences. arXiv; 2023. doi:10.48550/arXiv.1706.03741

17. Workshop B et al. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. arXiv; 2023. doi:10.48550/arXiv.2211.05100

18. Ostendorff M & Rehm G. Efficient Language Model Training through CrossLingual and Progressive Transfer Learning. arXiv; 2023. doi:10.48550/arXiv.2301.09626

19. Hoffmann J et al. Training Compute-Optimal Large Language Models. arXiv; 2022. doi:10.48550/arXiv.2203.15556

20. Paszke A et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library in Advances in Neural Information Processing Systems 32; 2019: 8024–8035.

21. Lhoest Q et al. Datasets: A Community Library for Natural Language Processing in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations; 2021. doi:10.18653/v1/2021.emnlp-demo.21

22. Gugger S et al. Accelerate: Training and inference at scale made simple, efficient and adaptable. https://github.com/huggingface/accelerate; 2022.

23. Rajbhandari S, Rasley J, Ruwase O & He Y. ZeRO: Memory optimizations Toward Training Trillion Parameter Models in SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, Atlanta; 2020: 1-16. doi: 10.1109/SC41405.2020.00024

24. Su Y & Collier N. Contrastive Search Is What You Need For Neural Text Generation. arXiv; 2023. doi:10.48550/arXiv.2210.14140

25. Su Y, Lan T, Wang Y, Yogatama D, Kong L & Collier N. A contrastive framework for neural text generation. Advances in Neural Information Processing Systems; 2022 (35): 21548-21561.

26. Zheng L et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. arXiv; 2023. doi:10.48550/arXiv.2306.05685

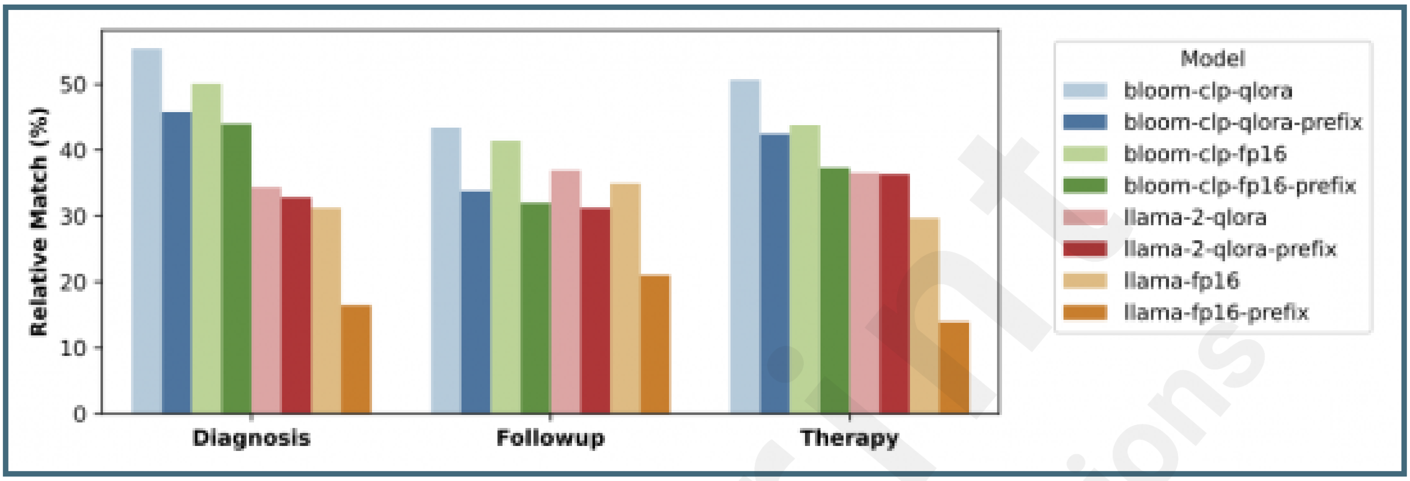
10) Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
CLP	Cross Lingual and Progressive Transfer Learning
EHR	Electronic Health Records
GDPR	General Data Protection Regulation
GPT	Generative Pretrained Transformer
HELM	Holistic Evaluation of Language Models
ICT	Incontext Learning
LLMs	Large Language Models
PHI	Protected Health Information
(Q)LoRA	(Quantized) Low Rank Adaptation
ZeRO	Zero Redundancy Optimizer

Supplementary Files

Figures

Fraction of reports in the test set where the models matched the extracted diagnosis, followup, and therapy recommendation of the doctor. All data are extracted using Claude-v2 (see section 2.5.1). For some models variants are trained using either full floating point 16-bit precision LoRA, indicated by the suffix ’fp16’ in the legend, or using NF4 integer precision QLoRA, indicated by the suffix ’qlora’. For all models we generate and evaluate two times: once using the standard prompt and once using a completion prefix (see section 2.4.1), indicated by the word ’prefix’ in the figure legend.


Relative Match (%)
Diagnosis
Followup
Therapy
Model
bloom-clp-qlora
bloom-clp-qlora-prefix
bloom-clp-fp16
bloom-clp-fp16-prefix
llama-2-qlora
llama-2-qlora-prefix
llama-fp16
llama-fp16-prefix