

HomeDOctor App: Integrating Medical Knowledge into GPT for Personal Health Counseling

Matic Zadobovšek
matic.zadobovsek@gmail.com
Faculty of Computer and
Information Science
Ljubljana, Slovenia

Primož Kocuvan
primoz.kocuvan@ijs.si
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

Matjaž Gams
matjaz.gams@ijs.si
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

Abstract

The escalating workload demands on healthcare professionals are leading to systemic overload, resulting in a decline in the efficiency of public health services. This situation necessitates the development of solutions that can alleviate the burden on physicians while ensuring comprehensive patient care. Recent advancements in generative artificial intelligence, particularly in the field of medicine, have demonstrated that large language models (LLMs) can outperform doctors in specific tasks, highlighting their potential as valuable tools for reducing the strain on healthcare providers. This study focuses on the development of the HomeDOctor application, which integrates additional validated medical knowledge into the GPT-4o LLM. The objective of this application and the enhanced LLM is to offer users reliable access to a medical chatbot capable of providing accurate and timely responses to health-related inquiries. The chatbot's behavior has been meticulously tested and refined in collaboration with a team of physicians. The findings of this research offer insights into the development of such systems and explore their potential application within the Slovenian healthcare system.

Keywords

large language models, GPT-4o, vector embeddings, vector databases, natural language processing, prompt engineering

1 Introduction

There are worldwide growing problems in the health sector due to an ageing population and a shortage of health experts [13, 17]. The field of Natural Language Processing (NLP) has recently seen an increase in the number of LLMs being customised for different domains [2]. Increasingly, we are also seeing their integration into the field of medicine, which is one of the core domains of today's society [14, 22, 11, 20]. Currently, OpenAI's GPT-4o is considered to be the most powerful LLM, which also performs best in the area of health questionnaires and other related tasks [16, 12, 1, 5].

Because of their advanced ability to understand natural language text, there are many potential applications [8, 3]. Patients can talk to the GPT, describe their symptoms in detail, include documents to past treatments and then get a friendly response in seconds. This is particularly useful when the patient is unable to access their personal doctor or simply wants to get feedback before taking further action. LLMs are trained on a huge amount of data, but there is the possibility of hallucination, especially

in less researched areas and specific issues [19]. As a result, it makes sense to think about augmenting the existing LLM with verifiable resources that can help to improve it. One of the most important issues regarding the use of LLMs relates to the aspect of secure use of health data. When using commercial LLMs, the data is passed to them and the user should be informed with appropriate disclaimers about the use and processing of the data they have entered. In an ideal situation, we would use our own LLM, running on a local server, and have full control over the implementation, but this raises different issues. Creating your own LLM is a time-consuming and costly process, so our research focused on extending the best LLM currently on the market, GPT-4o.

In our case, we focused on the Slovenian healthcare sector and the adaptation of the LLM to the needs of the average user. The aim was to develop an application (HomeDOctor) that would allow all Slovenian citizens to get medical help 24/7 without the need to contact a personal doctor. As a result, performance testing is crucial, as even the smallest errors can lead to serious consequences [8].

The core research hypothesis posits that the system, enhanced with additional modules integrated into a GPT, will provide more effective medical advice to the general Slovenian population compared to existing GPTs.

In section 2 we present all the datasets used to extend the LLM. Section 3 systematically describes the approach and technologies used to develop the software solution. Section 4 shows the results of the developed solution and an example of a user conversation with the improved LLM. Section 5 describes how performance was tested and how doctors helped to guide LLM to get the desired outcome. Section 6 presents concluding thoughts and possible improvements.

2 Datasets

To improve the knowledge of the existing ChatGPT-4o, data from verified sources was obtained. The addition of new data allows the GPT to answer questions using its existing knowledge, as well as to address a broader range of questions that require specialized knowledge. In this case, it is about integrating information on Slovenian healthcare [4].

2.1 Insieme Platform

The Insieme platform is the core building block of the dataset used [9]. The platform integrates hand-crafted expert-based knowledge that is accessible to users on all devices and contains basic information on Slovenian healthcare.

The information is organised hierarchically by medical specialty. By clicking (or visiting) on one of the branches, the user is redirected to a sub-page that provides an overview of the diseases and other services that belong to the selected field of medicine.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information Society 2024, 7–11 October 2024, Ljubljana, Slovenia

© 2024 Copyright held by the owner/author(s).

<https://doi.org/https://doi.org/10.70314/is.2024.chtm.8>

Each of the diseases is accompanied by a set of key data that allows the user to get everything they need in one place. This includes:

- basic information about the diagnosis,
- professional help (list of clinics in Slovenia),
- apps (where the user can upload their pictures, ask for opinion, etc.),
- associations in Slovenia,
- articles,
- forums,
- video content, and
- image content.

Figure 1 shows an example of one of the pages illustrating the display of the information for skin cancer.

All data have been compiled in collaboration with doctors and are therefore suitable for use as a primary source for the extension of the LLM. The aim is to enable users of the platform to quickly obtain information on Slovenian healthcare that they can understand.

2.2 Other resources

The Med.Over.Net Health Forum was used as one of the two remaining data sources. The forum is divided into different branches of medicine where users can create new conversations and ask questions within them. Each of the sub-forums has an assigned moderator (a doctor) who carefully reviews and answers the questions. All other forum users can also join the conversation. All data from the online forum was extracted using web scraping, whereby all extracted data was stored in a single PDF file.

The second source is the Home Doctor Encyclopaedia (The Home Doctor - Practical Medicine for Every Household), which is in the field of practical home medicine. It contains useful tips on how people can help themselves by applying helpful advice without the need to visit a doctor.

In the future, consideration will be given to adding new resources, in particular a better medical book suitable for Slovenian healthcare. For now, we have focused on the three data sources mentioned above.

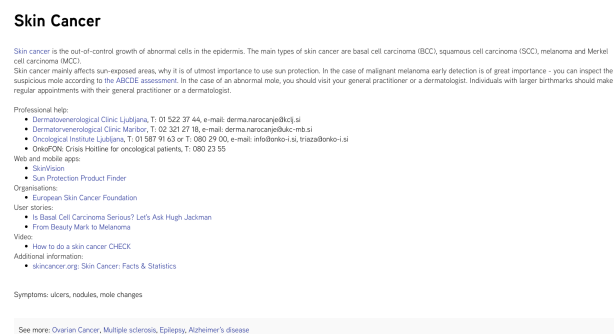


Figure 1: Image of the skin cancer disease subpage on the Insieme web platform.

3 Methodology

This section presents the design of a system that extends the LLM GPT-4 with acquired knowledge resources and outlines the architecture of the developed application.

3.1 Fine-tuning

One solution is fine-tuning, which allows a pre-trained LLM to add external data sources to a LLM that already has its own existing knowledge [10]. The idea is that instead of creating an LLM from scratch, an existing LLM is used and further adapted to the user's operational needs. Fine-tuning typically works on the principle of supervised learning, where the LLM needs to be given examples of the input and the expected output. With additional learning, the LLM is able to use the new data. This directly influences the change of parameters (weights) in the model and its performance. The problem arises in that, due to the large amount of parameters in modern LLMs, the whole learning process may be impractical due to the cost of training an outsourced LLM. Additionally, any incorporated data is static, meaning that the learning process must be repeated whenever adding new data. As a result, fine-tuning is not the best solution in cases where we know that the data will change, and we will be constantly adding new data.

3.2 Storing data

The data obtained by web scraping was stored in a vector database, which allows efficient management of the vector data [21].

The raw data obtained (e.g. PDF and Markdown files) are unsuitable for direct use with LLMs as they cannot be directly managed by the LLM. While documents can be added as attachments in the web interfaces of platforms such as ChatGPT (OpenAI's web LLM access), this is only useful in the context of a single conversation. In our case, we would like the LLM to have permanent access to information sources without the need for manual addition of documents by the user.

3.2.1 Vector embeddings. Each of the documents is converted into vector embeddings, which allow the text to be presented in a numerical notation that can be understood by a computer [6]. From this notation, the computer can understand the meaning of the text and determine the meaning between the words in the input. This is useful in many NLP tasks, e.g. search engines, sentiment analysis, recommendation systems and so on. We use vector embeddings to be able to extract information relevant to the corresponding question based on user input.

Modern LLMs have a limitation in the form of a context window, which determines how much text they can process within a single user input. Because of this limitation, we need to be careful what additional information is sent to the LLM, as we could quickly exceed the context window by sending all data sources. E.g. if the user's question is about skin cancer, we only want to get information about that specific disease. As a consequence, we have divided the text into chunks, where each chunk contains data for only one of the diseases. Similar care has been taken with the encyclopaedia and the Med.Over.Net online forum. Meaningful paragraphs have been grouped together to form one chunk of text. Thus, only the key chunk that is most likely to contain the answer to the question asked will be provided to the LLM.

For each of the chunks, a vector embedding has to be created using an appropriate model (e.g. text-embedding-ada-002 from OpenAI). The vector databases (e.g. Redis) have to be used to store the resulting vector embeddings properly.

3.2.2 Vector databases. The vector database allows the storage of unstructured data and fast retrieval due to efficient indexing of the data. In our case, we use them to store individual chunks and their associated vector embeddings [7].

Over all stored chunks, vector search is enabled, which means that the chunks whose vector embeddings most closely match the query vector are returned. The query vector represents the user input, which is converted into vector format. Cosine similarity, Euclidean distance, inner product and other metrics can be used to measure similarity between vectors.

The configuration used to retrieve the chunks can be modified: e.g. specifying the required similarity threshold and the number of chunks retrieved.

3.3 RAG

An alternative solution is Retrieval-Augmented Generation (RAG), where the LLM calls an external database containing all our data when the user provides a question [15]. Relevant data is retrieved from there and passed directly to the LLM, which uses this data in the generation of the answer. In this case, the basic architecture of the LLM used remains completely unchanged, as it accesses a separate building block to retrieve the data. It is a cheaper solution that allows dynamic data extraction. Data can be deleted, modified and newly added to the database at any time.

The Retriever is responsible for retrieving relevant pieces of text from the vector database. The user question is converted into vector embedding, and then the most similar chunks are retrieved to help guide the LLM to the correct answer. A merged query is then created containing the original user question, the extracted chunks of text and any other system instructions given. In the final step, the LLM generation produces an answer based on the query, which is passed to the user.

Compared to fine-tuning, RAG allows the system to change continuously (adding new knowledge sources), makes the operation more understandable (we can check which pieces of text have been passed to the LLM) and reduces the possibility of hallucination (verified pieces of text from the selected domain are added). The style of writing the answers cannot be changed directly, as the parameters of the original model remain completely unchanged, but we can help by writing system instructions. System instructions further guide the operation of the chatbot and play a very important role in achieving a user-friendly behaviour of the system.

All of these features make RAG a suitable choice for enriching the work of an existing LLM with validated information in the chosen field.

The schematic design of the RAG system can be seen in the figure 2 below.

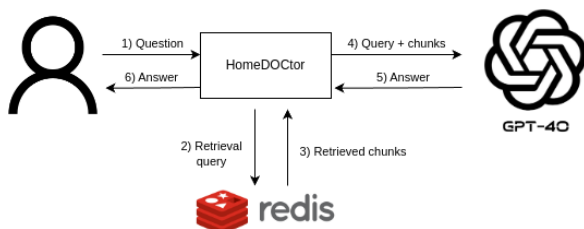


Figure 2: Demonstration of how the RAG system works.

3.4 Application architecture and used technologies

A brief overview of the technologies used to develop the software solution is presented below.

3.4.1 Flutter. The Flutter framework was used to develop the front-end part of the system. The reason for using it is that it allows us to develop applications that work on all platforms by writing unified code in the Dart programming language. It then gets compiled into code for the dedicated platform. This way, the application works on web browsers as well as in the form of a mobile application. Otherwise, we would have to use multiple programming languages to ensure support on multiple devices, which would slow down development.

3.4.2 Redis. The Redis database was used to store all additional knowledge resources. It is a NoSQL database that stores data in key-value format and also works as a vector database. The data is stored in memory, which helps significantly with data retrieval and overall application responsiveness.

3.4.3 LangChain. The LangChain framework offers a number of functionalities that facilitate the development of applications that involve working with LLMs [18]. It allows us to easily connect to a vector database, retrieve documents, call arbitrary LLMs, modify system instructions and so on.

3.4.4 FastAPI. The FastAPI framework allows the creation of RESTful APIs that can be accessed by the HomeDOCTOR application. It uses Uvicorn for the server implementation and allows processing of requests received from users. These requests are related to the processing of the received questions and the execution of all the necessary tasks to obtain the appropriate response from the LLM.

3.4.5 Docker. The entire application was packaged in a Docker container to facilitate development and deployment to the server. All the necessary configuration is included in the container, which runs on the system without the need to install the rest of the programming libraries and tools.

3.4.6 Firebase. The Firebase platform offers a number of useful functionalities such as user authentication, application usage analytics, cloud storage and a real-time database (NoSQL cloud database, which in our case is used to store the necessary user data).

4 Results

The HomeDOCTOR app allows users to access a chatbot that is constantly available to answer their questions. Users can access the app on their browsers or via a mobile app. The user interface allows users to view the history of the conversation, ask questions, include their own documents as an attachment (e.g. PDF documents of previous exams), get a PDF summary of the conversation and additional information in three side windows. By clicking on the "End session" button at the bottom of the screen, all data is deleted, and the user is redirected to the main page where they can start a new conversation with the chatbot.

Initially, HomeDOCTOR determines what kind of conversation is being had. In the case of a simple chat, no additional information is displayed. However, if the system detects that the conversation is about the user's health, additional information is displayed in three side windows. This additional information is dynamically updated according to the progress of the user's

conversation with the chatbot. The first window gives a list of possible medical conditions, the second window gives advice on how the user could help himself, and the third window tells whether a visit to the doctor would be necessary. This way, the user can receive a comprehensive overview of the conversation so far and get an opinion that adapts to the course of the conversation.

Figure 3 below shows the user interface of the HomeDOCTOR application. We can see the user’s conversation with the chatbot in Slovenian and the display of additional information in the three side windows along the right edge. When answering questions, GPT-4o uses its existing knowledge, incorporating as often as possible additional sources of knowledge (especially from the Insieme platform).

Using the app, users get instant help from a chatbot. The chatbot first tries to collect sufficient data before proceeding to further advice. In the course of the treatment, the user is advised on different forms of help (websites, questionnaires, forums, apps, visits to pharmacies, etc.). A visit to the doctor is only advised in urgent cases, when HomeDOCTOR considers that the situation is too complicated or further medical treatment is needed.

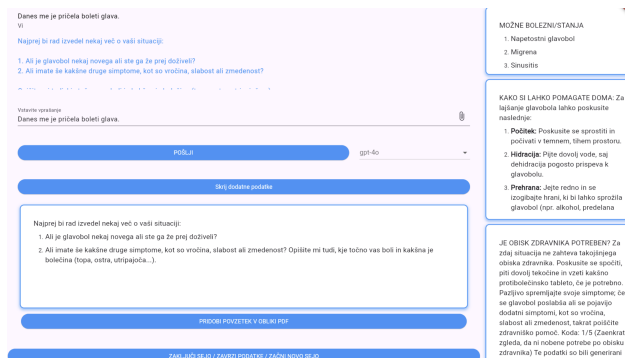


Figure 3: An example of a user talking to the HomeDOCTOR chatbot. Additional information is visible on the side.

An example of a conversation from a user who consulted the developed chatbot is shown below. We can see the user’s questions and all the corresponding answers.

- **User:** I have a slight headache and a bit of a sore throat today.
- **HomeDOCTOR:** I understand. Let’s gather some more information to better assist you.
 1. How long have you been experiencing these symptoms?
 2. Do you have any other symptoms, such as fever, cough, or body aches?
- **User:** I’ve been feeling this way for about two days now. It started with just a mild sore throat, and the headache came yesterday.
- **HomeDOCTOR:** Thanks for the information.
 1. How severe are your headache and sore throat on a scale from 1 to 10?
 2. Have you noticed any other symptoms, like nasal congestion, fatigue, or body aches?
- **User:** I’d say my headache is about a 4 out of 10. It’s more of a dull ache, not too intense, but definitely noticeable. The sore throat is a bit worse, probably a 6 out of 10.
- **HomeDOCTOR:** Thank you for the details.
 1. Have you taken any over-the-counter medications or home remedies for relief? If so, have they helped?

2. Is there anything that seems to make your symptoms better or worse, such as certain foods, drinks, or activities?

- **User:** I tried drinking some warm tea with honey, which soothed my throat for a while. My headache seems to get a bit worse when I’m looking at screens for too long.
- **HomeDOCTOR:** Thank you for the additional information. It sounds like you might be experiencing symptoms of a common cold or another mild infection. Here are a few recommendations:
 1. Rest and Hydration: Ensure you get plenty of rest and stay well-hydrated.
 2. Warm Teas: Continue drinking warm teas with honey to soothe your throat.
 3. Limit Screen Time: Try to reduce your screen time to help with your headache.
 4. Steam Inhalation: Consider steam inhalation to relieve any nasal congestion you might have.
 For additional relief, you can visit a pharmacy for over-the-counter options. You can find more information about medications at <https://mediately.co/si> and <http://www.cbz.si>. If your symptoms worsen or persist for more than a week, it’s advisable to consult a healthcare professional.
- **User:** Thank you for the advice. Where could I get more information?
- **HomeDOCTOR:** You’re welcome! For more information on common conditions and treatments, you can check these resources:
 1. Med.Over.net: <https://med.over.net>
 2. Ars Pharme - Zdravstveni nasveti: <https://www.arspharmae.com>
 3. Triglav Zdravstveni Nasveti: <http://www.triglavzdravje.si>
 If you need more personalized advice, consider visiting a pharmacist or consulting with a healthcare professional.

5 Evaluation

The full operation of the HomeDOCTOR app and chatbot was carried out by the authors and medical experts. After consulting with doctors and identifying the key performance points, iterative and collaborative development followed, gradually introducing changes and improving the performance of the chatbot.

Doctors have contributed significantly to improving the app through rigorous testing, ensuring that the user can get the best possible results to their questions. We have been careful to address situations where a doctor’s visit is not necessary. In many cases, non-urgent situations only add to the workload of doctors. In our system, we have tried to avoid this as much as possible by trying to refer the user to other resources that would be good and useful enough for them, and only as a last resort recommending a visit to the doctor. Of course, a visit to the doctor should not always be avoided, so it is crucial that the system is able to correctly identify scenarios where this is really necessary. These are mainly cases where additional diagnostics are needed, which are difficult for the LLM to perform on its own. Doctors have been particularly helpful in the project by further adapting the system instructions, which detail how to deal with all scenarios.

We compared HomeDOCTOR with other large language models on the market, but focused the most on comparing it with the regular GPT-4o model, which our system uses for its basic operation. The main differences are in the flow of the conversation, as HomeDOCTOR is customised by the system instructions to follow specific scenarios and ask questions that are key for

providing further advice. Thus, by simulating a conversation of a user who starts the conversation by listing symptoms, a comparison between the two systems can be made. According to the clinicians' evaluations, HomeDOCTOR appeared to be better at asking sub-questions and guiding the user through the treatment. A further important difference can be seen in the display of videos and images, as HomeDOCTOR can display them right within the app, whereas GPT-4o creates fictional images or often provides links to non-existent videos. HomeDOCTOR also often gives practical advice on how users can help themselves at home and carefully evaluates when it is really necessary to redirect the user to professional help.

Testing of the system has demonstrated the added value of the included knowledge sources, though formal validation tests are still pending. Nevertheless, the scientific hypothesis that superior performance can be achieved compared to the original GPT is increasingly supported by the evidence. Through iterative testing by all participants, the system has now reached a level of maturity suitable for deployment in a production environment. Test users will be able to provide feedback and suggest corrections via a form, which will further refine the chatbot to better meet the needs and preferences of the average user in Slovenia. This will display the data sources used to the test users, who will be able to indicate whether they believe that the relevant documents have been extracted from the vector database.

6 Conclusion

In this study, we have shown how verified information can be incorporated into one of the existing LLMs. It was an extension of GPT-4o, which currently achieves the best results, and has been further adapted to the needs of Slovenian public health in the context of the development of the HomeDOCTOR application.

Comprehensive testing has been carried out with a team of doctors, which has helped to ensure that users are provided with relevant and understandable information. HomeDOCTOR initially gathers information by asking questions, and only then advises on further action. External resources and self-help methods are prioritised, as the intention is that referral to a doctor should only be made in urgent situations.

Possible improvements are in the use of one of the open source LLMs, which would negate the need for external access to OpenAI's LLMs. Currently, in order to run the HomeDOCTOR application, an API call is sent for each question asked, for which a fee is paid to OpenAI. This is also problematic from a data protection point of view as it involves working with confidential user data which should not be passed on (although you are default opt-out for OpenAI API). Currently, this is taken care of with disclaimers and warnings that the user has to agree to before using the application. At the same time, the use of open source models would bring additional problems, as the appropriate infrastructure would have to be established. Open source models also typically perform worse than e.g. GPT-4o. As a result, we have chosen to use GPT-4o, as we cannot afford worse outcomes due to the criticality of the medical domain.

HomeDOCTOR could also be adapted to meet the healthcare needs of countries abroad, but this would require ensuring that the relevant data is obtained in accordance with their national guidelines and security laws. The behaviour of the chatbot could remain largely the same, as GPT-4o can by itself convert between many languages. The key component would thus be to obtain all the necessary country-specific information that we want to

make available to users (information on clinics, apps, articles, video content in that language).

In the future, work will be carried out on improving the chatbot, taking into account the opinions of external users, who in this case are also our target group. This research has shown that generative artificial intelligence has a great potential application in the field of medicine and could make a significant contribution to relieving the burden on the healthcare system.

Acknowledgements

We thank members of the Department of Intelligent Systems and medical experts for providing info and testing the system. This project is funded by the European Union under Horizon Europe (project ChatMED grant agreement ID: 101159214).

References

- [1] Josh Achiam et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- [2] Jan Clusmann et al. 2023. The future landscape of large language models in medicine. *Communications medicine*, 3, 1, 141.
- [3] Matjaž Gams and Tine Kolenik. 2021. Relations between electronics, artificial intelligence and information society through information society rules. *Electronics*, 10, 4, 514.
- [4] Matjaž Gams, Maj Smerkol, Primož Kocuvan, and Matic Zadobovšek. 2024. Developing a medical chatbot: integrating medical knowledge into gpt for healthcare applications. In *Intelligent Environments 2024: Combined Proceedings of Workshops and Demos & Videos Session*. IOS Press, 88–97.
- [5] Aidan Gilson, Conrad W Safranek, Thomas Huang, Vimig Socrates, Ling Chi, Richard Andrew Taylor, David Chartash, et al. 2023. How does chatgpt perform on the united states medical licensing examination (usmle)? the implications of large language models for medical education and knowledge assessment. *JMIR medical education*, 9, 1, e45312.
- [6] Martin Grohe. 2020. Word2vec, node2vec, graph2vec, x2vec: towards a theory of vector embeddings of structured data. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, 1–16.
- [7] Yikun Han, Chunjiang Liu, and Pengfei Wang. 2023. A comprehensive survey on vector database: storage and retrieval technique, challenge. *arXiv preprint arXiv:2310.11703*.
- [8] Yining Huang, Keke Tang, and Meilian Chen. 2024. A comprehensive survey on evaluating large language model applications in the medical industry. *arXiv preprint arXiv:2404.15777*.
- [9] Jožef Stefan Institute. 2024. Insieme platform. (Aug. 2024). Retrieved August 26, 2024 from <https://ise-emh.eu>.
- [10] Jieh-Sheng Lee and Jieh Hsiang. 2020. Patent claim generation by fine-tuning openai gpt-2. *World Patent Information*, 62, 101983.
- [11] Peter Lee, Carey Goldberg, and Isaac Kohane. 2023. *The AI revolution in medicine: GPT-4 and beyond*. Pearson.
- [12] Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.
- [13] World Health Organization et al. 2022. Health and care workforce in europe: time to act.
- [14] Carl Preiksaitis and Christian Rose. 2023. Opportunities, challenges, and future directions of generative artificial intelligence in medical education: scoping review. *JMIR medical education*, 9, e48785.
- [15] Mujahid Ali Quidwai and Alessandro Lagana. 2024. A rag chatbot for precision medicine of multiple myeloma. *medRxiv*, 2024–03.
- [16] Nicholas R Rydzewski, Deepak Dinakaran, Shuang G Zhao, Eytan Ruppim, Baris Turkbey, Deborah E Citrin, and Krishnan R Patel. 2024. Comparative evaluation of llms in clinical oncology. *Nejm Ai*, 1, 5, A10a2300151.
- [17] Tony Smith. 1991. Medicine in europe. european health challenges. *BMJ: British Medical Journal*, 303, 6814, 1395.
- [18] Oguzhan Topsakal and Tahir Cetin Akinci. 2023. Creating large language model applications utilizing langchain: a primer on developing llm apps fast. In *International Conference on Applied Engineering and Natural Sciences* number 1. Vol. 1, 1050–1056.
- [19] Ashish Vaswani. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.
- [20] Ethan Waisberg, Joshua Ong, Mouayad Masalkhi, Sharif Amit Kamran, Nasif Zaman, Prithul Sarker, Andrew G Lee, and Alireza Tavakkoli. 2023. Gpt-4: a new era of artificial intelligence in medicine. *Irish Journal of Medical Science (1971-)*, 192, 6, 3197–3200.
- [21] Matic Zadobovšek, Primož Kocuvan, and Matjaž Gams. 2023. Vpeljava virtualnega asistenta chatgpt v medicinsko platformo = implementation of a virtual assistant chatgpt into the medical platform. In *Bibliografija*: str. 31. Institut „Jožef Stefan“, 28–31. http://library.ijs.si/Stacks/Proceedings/InformationSociety/2023/IS2023-Volume_A%20-%20SKUI.pdf.

- [22] Peng Zhang and Maged N Kamel Boulos. 2023. Generative ai in medicine and healthcare: promises, opportunities and challenges. *Future Internet*, 15, 9, 286.