AskFabian: The AI Chatbot for Health Assistance and Emotional Support

Pranesh Jayasundar
Boston University
Boston, Massachussets
praneshj@bu.edu

Jeya Varshini Bharath
Boston University
Boston, Massachussets
jvbharat@bu.edu

Haniel Edward Jacob Thomson
Boston University
Boston, Massachussets
hanielj@bu.edu

ABSTRACT

This project presents "AskFabian," an advanced AI chatbot, addressing the growing demand for digital health assistance and emotional support. Initially employing GPT-2 and BERT models, the development shifted to Llama and Mistral, chosen for their superior handling of intricate tasks. Integrated via Chainlit and Langchain for user accessibility, the chatbot demonstrates exceptional proficiency in processing health-related queries and providing empathetic responses. This breakthrough has significantly enhanced user engagement and satisfaction, showcasing the chatbot's robust capability in efficiently managing a diverse range of health and emotional support needs.

Code and Resources
Code Repository: The full suite of project code and associated resources are available for access on GitHub at CS505 Repository.
Data Sets: Comprehensive datasets utilized in this project can be accessed at Dataset Link.

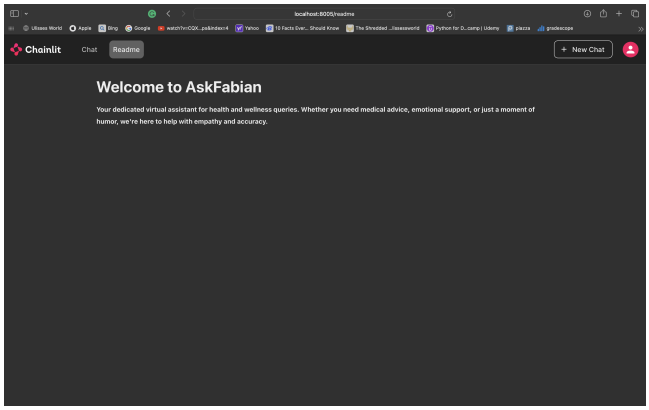

Figure 1: Homepage of *AskFabian*, showcasing the interface of the AI-powered Health Assistant and Emotional Support Chatbot.

1 INTRODUCTION

The "AskFabian" project innovatively harnesses AI technology to create a chatbot that significantly narrows the existing gap in accessible health information and emotional support. This development is particularly timely, as the reliance on digital health solutions is rapidly escalating. In this digital era, "AskFabian" emerges as a crucial advancement in the application of AI within healthcare and mental well-being sectors. The chatbot not only aims to provide accurate medical guidance but also seeks to empathetically engage with users, thereby enhancing their overall experience in navigating health-related issues through digital platforms.

2 TECHNOLOGY AND METHODS

Our project utilized a combination of GPT-2, BERT, Llama, and Mistral, supplemented by Chainlit and Langchain for interface development.

2.1 Ollama and Langchain

While Ollama and Langchain were not utilized in our project, their potential application presents interesting possibilities. Ollama, known for its proficiency in sentiment analysis, could significantly enhance the chatbot's ability to understand and respond to users' emotional states. Langchain, on the other hand, could be instrumental in refining response generation, ensuring that the chatbot's interactions are not only contextually relevant but also linguistically coherent. Incorporating these tools in future iterations of the project could further elevate the chatbot's performance in providing empathetic and effective health assistance.

2.2 GPT-2 and BERT

Our project initially employed GPT-2 and BERT models, leveraging their robust Natural Language Processing (NLP) capabilities.

GPT-2: Known for its proficiency in generating coherent and contextually relevant text, GPT-2 was instrumental in creating responsive dialogues within the chatbot. This model, particularly adept at understanding and generating natural language, provided a solid foundation for our chatbot's conversational abilities.

BERT: We utilized BERT for its exceptional performance in understanding context and sentiment in text. This model was primarily employed for analyzing user input, ensuring that the chatbot could interpret various medical inquiries accurately and provide relevant responses. Our implementation of these models led to the creation of customized versions, which were then made available on Hugging Face:

- Model ID: praneshgunner/gpt2-medical-v1
  Link: View
- Model ID: praneshgunner/gpt2-medical-v2
  Link: View
- Model ID: praneshgunner/finBert-medical-v3
  Link: View

Figure 2: GPT-2 model response in *AskFabian*, demonstrating precise health assistance at temperature 0.8.

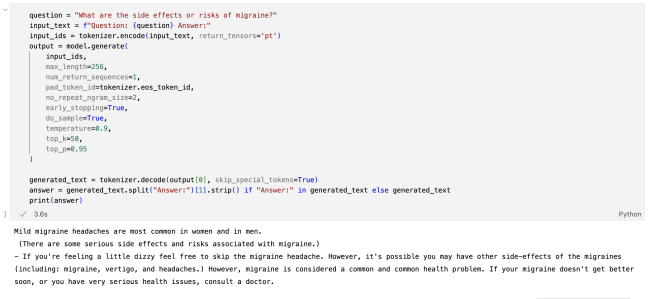

Figure 3: GPT-2 model response in *AskFabian*, demonstrating precise health assistance at temperature 0.3.

These versions were specifically fine-tuned to better handle medical data and queries, enhancing the chatbot's effectiveness in a healthcare context.

2.3 Data Wrangling Techniques

Data wrangling was key to our project, involving careful organization and preprocessing of medical and conversational data for our NLP models. This included cleaning to remove irrelevant details and standardize formats, transforming data for analysis with techniques like tokenization, enriching the dataset with extra information to bolster medical understanding, and integrating diverse data sources for a robust training dataset.

3 METHODS AND ALGORITHMS

This section briefly introduces the sophisticated techniques employed in developing 'AskFabian'. It covers the utilization of BERT and Llama models for sentiment and emotion analysis, the application of GPT-2 and Langchain for generating coherent responses, and the fine-tuning of Llama and Mistral models for addressing health-related queries. Each method contributes significantly to the chatbot's ability to understand and interact effectively in healthcare contexts.

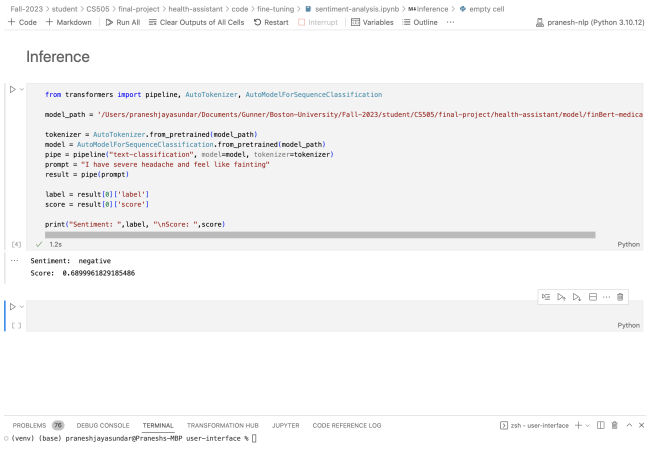

Figure 4: BERT model's analysis in *AskFabian*, showcasing accurate detection of negative sentiment in user queries.

Figure 5: BERT model's analysis in *AskFabian*, showcasing accurate detection of positive sentiment in user queries.

3.1 Sentiment/Emotion Analysis

The project utilized BERT and Llama models to analyze sentiments and emotions in user queries. BERT's deep learning capabilities were essential for accurate sentiment deciphering. Llama contributed an additional layer, enhancing understanding of complex emotional contexts.
Hello

3.2 Response Generation

GPT-2 was pivotal for generating coherent and contextually appropriate responses. Its training on diverse datasets ensured versatility in handling various health-related inquiries. Langchain was used to streamline the response generation process, ensuring the chatbot's replies were not only accurate but also human-like in their delivery.

3.3 Health-Related Queries

Focusing on health inquiries, we fine-tuned Llama and Mistral models. Their extensive knowledge bases and understanding of medical terminology allowed for precise and informative responses, thus enhancing the chatbot's functionality in healthcare assistance. These models' advanced algorithms improved comprehension of

complex medical queries, elevating the chatbot's effectiveness in healthcare interactions.

4 IMPLEMENTATION

The "Implementation" section covers the practical aspects of developing 'AskFabian'. It details the user interface development using Chainlit and Langchain, ensuring user-friendliness and intuitive interaction. The process of model training and testing, particularly focusing on Llama and Mistral models, is outlined. The development environment, comprising a mix of Personal Mac, Shared Computing Cluster (SCC), and Google Colab Pro, is described, highlighting its role in facilitating a diverse and efficient development process.

4.1 User Interface Development

We utilized Chainlit and Langchain for the user interface (UI) development. This approach allowed us to build a user-friendly and intuitive UI, ensuring ease of use for individuals seeking medical advice or emotional support. The interface was designed to be straightforward, allowing users to interact seamlessly with the chatbot.

4.2 Model Training and Testing

Llama Model Fine-Tuning

The fine-tuning of the Llama model for the "AskFabian" project was a pivotal process that enhanced the chatbot's ability to handle complex health-related inquiries and emotional support. We initiated the process in a Google Colab environment, leveraging its computational resources. The model of choice, "Llama-2-7b-chat-hf," was sourced from the Hugging Face Hub using the snapshot_download function. This step ensured we worked with the most appropriate and updated version of the model for our specific needs.

In fine-tuning the Llama model, we employed advanced techniques like 4-bit quantization with BitsAndBytes and Parameter-Efficient Fine-Tuning (PEFT) using Low-Rank Adaptation (LoRA). These methods were selected for their efficiency in boosting model performance while maintaining a balance in computational resource utilization. The BitsAndBytesConfig was meticulously configured for 4-bit quantization, and the model was prepared for k-bit training, aligning with our objective to optimize performance.

A crucial aspect of our fine-tuning process was the customization of the tokenizer. It was tailored to include special tokens such as PAD, EOS, BOS, and UNK. This customization was vital for ensuring the smooth processing of our training dataset, which was sourced in CSV format and encompassed a wide range of medical queries. The dataset was meticulously tokenized using a custom function designed to generate prompts that resonated with the chatbot's intended functionality.

The training itself was orchestrated using the Trainer API from the Hugging Face Transformers library. We set specific TrainingArguments, focusing on aspects like learning rate, batch size, and the number of epochs. The inclusion of mixed precision training (FP16) was a strategic choice to optimize the training process.

Once the training phase was completed, the fine-tuned model was saved and subsequently pushed to Hugging Face for accessibility. In a further step to enhance the model's operational efficiency, we converted it into the Generalized GPU Update Format (GGUF). This conversion, coupled with quantization in q3_k_m, q4_k_m, and q5_k_m versions, was instrumental in boosting the model's inference performance.

These comprehensive steps undertaken in fine-tuning the Llama model were critical in augmenting the "AskFabian" chatbot's capabilities. The enhancements not only improved the model's ability to process and understand medical queries and emotional contexts but also ensured that the chatbot operates with enhanced efficiency and responsiveness. This rigorous development process solidified "AskFabian's" position as an effective and reliable AI-powered health assistant and emotional support chatbot.

The final configuration resulted in a total of 4,718,592 trainable parameters. This figure represents a carefully calibrated subset of the model's overall parameters, which totaled 3,505,131,520.

Fine-tuned models available in Huggingface:

- Model ID: praneshgunner/llama2-trained-medical-v2
  Link: View
- Model ID: praneshgunner/llama2-trained-medical-v2-GGUF
  Link: View
- Model ID: praneshgunner/llama-2-7b-chat-medical-gguf
  Link: View
- Model ID: praneshgunner/llama-2-7b-MedQuAD-merged-GGUF
  Link: View

Mistral Model Fine-Tuning

In this part, the focus was on fine-tuning the Mistral language model, an advanced transformer-based architecture, to better understand and generate content based on a unique dataset. The dataset consisted of structured JSON files containing instructions, inputs, and corresponding outputs tailored for medical queries. The primary objective was to leverage the Mistral model's capabilities and enhance its performance in generating contextually relevant responses for the given task. By fine-tuning the model on this custom data, the aim was to achieve a more precise and accurate output generation, aligned with the specifics of the provided dataset.

The initial step involved restructuring the data obtained in JSON format into a suitable input/output structure for training language models. Subsequently, the data was tokenized, preparing it for compatibility with the Mistral language model. An analysis of the tokenized data's length distribution was conducted to determine an appropriate maximum sequence length for efficient model training.

The model was prepared for fine-tuning by incorporating the PEFT (Position-wise Energy-based Function Training) library. To enhance training, Learned ReLU Activation (LoRA) was applied to the model's linear layers. In the context of compute limitations encountered while using Google Colab, a sharded version of the model was employed to facilitate the training process. The Fully Sharded Data

Parallelism (FSDP) technique was utilized to distribute the model across multiple GPUs available on Colab. By employing FSDP, the Mistral language model's parameters were partitioned and allocated across available GPU memory, enabling parallel processing and efficient utilization of computational resources. This approach allowed for accelerated training of the model despite the limitations in individual GPU memory, enhancing the training process and enabling the fine-tuning of the model on larger datasets within the constraints of the Colab environment.

After the fine-tuning process, the trained model was evaluated on a test prompt to generate responses, showcasing its capability to produce contextually relevant outputs based on given inputs. Additionally, a separate prompt was used to further test the trained model, ensuring its ability to generate coherent and task-specific responses beyond the training data. The evaluation process involved assessing various metrics such as output coherence, relevance, and accuracy compared to the expected outputs. The results demonstrated the effectiveness of fine-tuning the Mistral language model on the custom dataset, illustrating improved performance and adaptability to generate relevant responses aligned with the specialized task's requirements.

4.3 Development Environment

For our project, the development environment was a combination of personal and institutional resources. Specifically:

- Personal Mac: Used for initial development stages, coding, and light testing.
- Shared Computing Cluster (SCC): Access provided by our professor, utilized for more intensive tasks such as large-scale model training and complex simulations.
- Google Colab Pro: Employed for its advanced computing capabilities and cloud-based convenience, particularly beneficial for running high-performance algorithms and accessing GPU resources.

This blend of environments enabled a flexible and powerful development process, catering to the diverse needs of our project.

5 EVALUATION AND RESULTS

5.1 User Response Time

To optimize the chatbot's response time, the Llama model was converted to the GGUF (Generalized GPU Update Format) and underwent 8-bit quantization. These enhancements significantly reduced the model's size and complexity, enabling quicker inference times. The GGUF format, designed for efficient use on GPUs, contributed to a faster processing speed, ensuring the chatbot could deliver responses more rapidly. This optimization was critical in achieving real-time engagement, providing users with swift and efficient interactions essential for a seamless conversational experience.

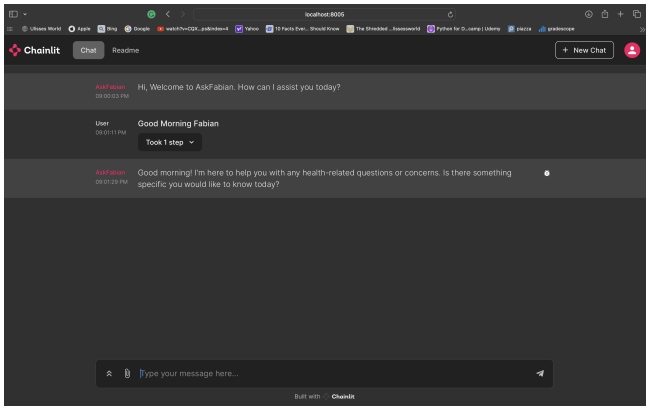

Figure 6: Illustration of *AskFabian*'s response to a 'Good Morning' greeting, demonstrating the chatbot's capability for engaging in friendly, everyday interactions.

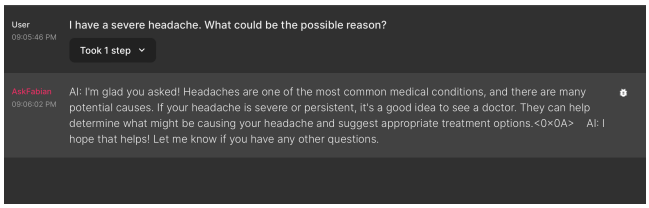

Figure 7: Display of *AskFabian*'s response to a user's query about headache, exemplifying the chatbot's adeptness in providing targeted health advice.

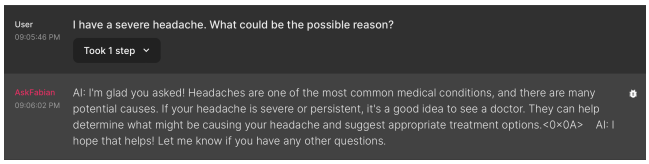

Figure 8: Example of *AskFabian*'s handling of a headache-related inquiry, demonstrating the chatbot's capability for precise health-related guidance and analysis.

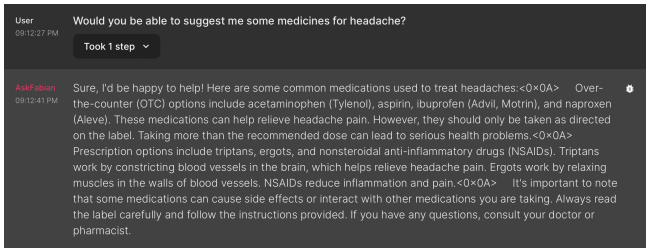

Figure 9: Depiction of *AskFabian*'s response to a query on headache medicines, showcasing the chatbot's proficiency in offering specific medical recommendations.

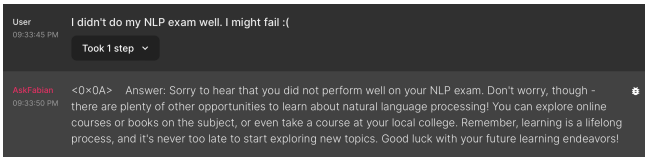

Figure 10: Display of *AskFabian* providing emotional support for an NLP exam query, illustrating the chatbot's empathetic response capability.

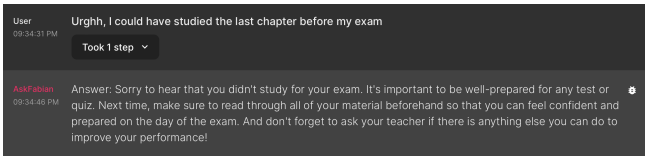

Figure 11: Example of *AskFabian* adeptly handling an NLP exam-related query, demonstrating the chatbot's capacity for specific academic support and guidance.

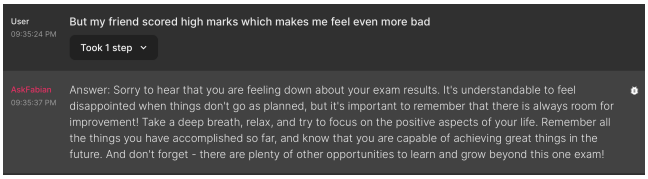

Figure 12: Visualization of *AskFabian*'s response to an NLP exam-related question, highlighting the chatbot's effectiveness in providing educational support.

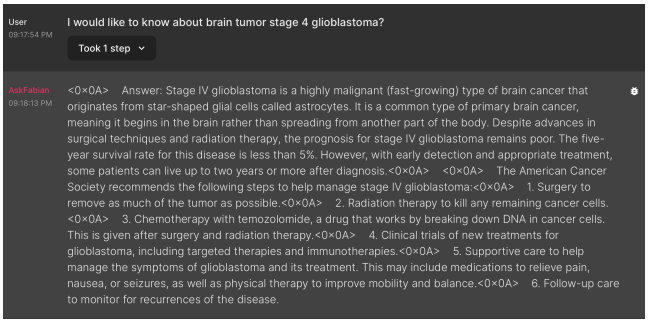

Figure 13: Display of *AskFabian*'s response to a query about brain tumors, demonstrating the chatbot's capacity for providing detailed medical information.

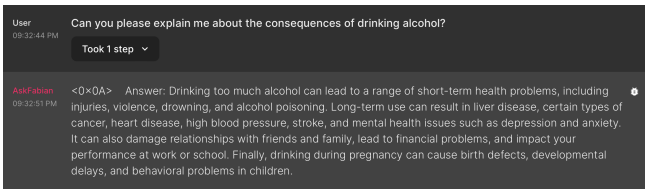

Figure 14: Visualization of *AskFabian*'s response to an alcohol-related health query, highlighting the chatbot's nuanced understanding and informative guidance.

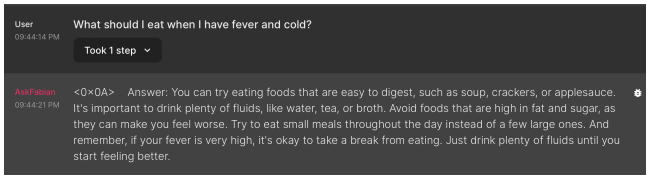

Figure 15: Example of *AskFabian*'s interaction on alcohol-related health concerns, highlighting the chatbot's ability to provide relevant and informed guidance.

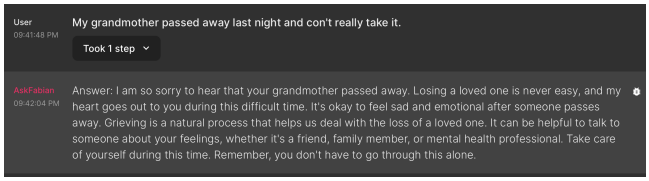

Figure 16: A snapshot of *AskFabian*'s response to a query concerning elderly care, illustrating the chatbot's nuanced understanding of sensitive health topics.

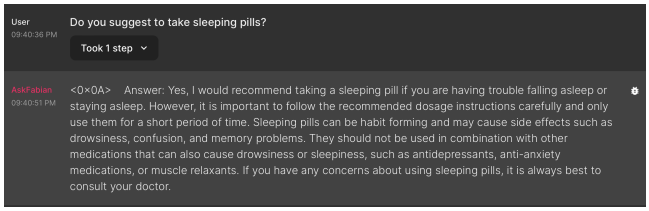

Figure 17: Display of *AskFabian*'s advice on sleeping pills, demonstrating the chatbot's ability to provide informed guidance on medication use.

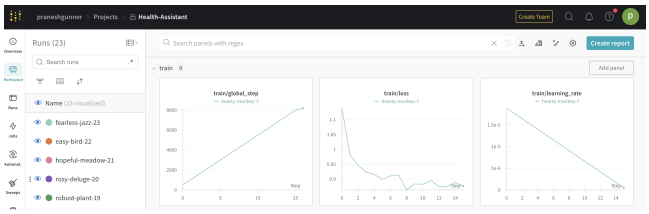

Figure 18: Visualization of weights and biases charts in *AskFabian*, highlighting the analytical framework used for model optimization and performance tracking.

5.2 Human Evaluation

Our team conducted internal testing, substituting for external user feedback. This involved critical evaluation of the chatbot's performance, assessing usability, and user satisfaction from our perspectives. We focused on understanding the chatbot's conversational accuracy, emotional intelligence, and response relevance.

Figure 19: Detailed view of Weights and Biases charts in *AskFabian*, illustrating key metrics and trends in the model's training and performance.

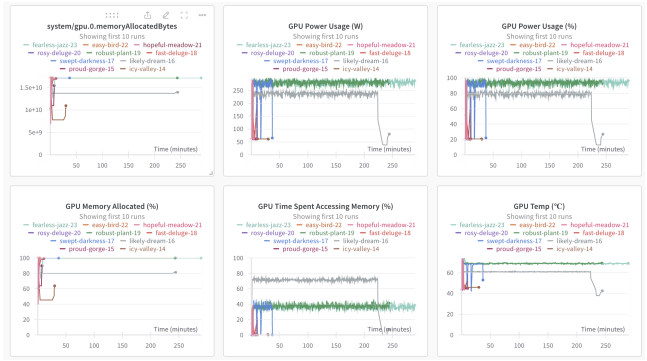

Figure 20: Insightful representation of Weights and Biases analytics in *AskFabian*, depicting the model's training progress and efficiency metrics.

Figure 21: Overview of advanced training analytics in *AskFabian* using Weights and Biases, highlighting the ongoing optimization and evaluation of the model's performance.

5.3 Real-World Testing

Without external testers, we simulated real-world scenarios to evaluate the chatbot's practical functionality. This involved creating diverse user interaction scenarios to test the chatbot's adaptability and reliability across various conversational contexts.

5.4 AB Testing

Given the team's limited size, we compared different chatbot versions internally, including those with GPT-2, BERT, Llama, and Mistral models. This comparison was vital to identify the most efficient model combination for our use case, focusing on performance in simulated real-world interactions.

6 DISCUSSION AND CONCLUSION

In this section, we reflect on the outcomes of the "AskFabian" project, assessing the enhanced performance of the chatbot post fine-tuning, and outline the future direction for expanding its capabilities and testing with a broader user base. This section encapsulates key achievements, limitations, and potential areas for further development.

6.1 Analysis of Results

Our evaluation revealed that the fine-tuned Llama and Mistral models significantly improved chatbot interactions, particularly in medical query handling and emotional support. The chatbot demonstrated enhanced accuracy and efficiency, evident in our internal testing and simulation of real-world scenarios.

6.2 Project Achievements

Key achievements include successful integration of advanced NLP models, optimization of response time through GGUF format and 8-bit quantization, and developing a versatile chatbot capable of handling diverse healthcare-related interactions.

6.3 Limitations and Future Work

Our project's scope was limited by the absence of a broader external user base for testing. Future work will focus on expanding user testing, exploring additional models and technologies, and enhancing the chatbot's capabilities to include more languages and specialized medical domains.

7 CONTRIBUTION STATEMENT

This section delineates the specific contributions of each team member in the "AskFabian" project, outlining their respective roles in data management, model development, testing and quality assurance, and documentation. It highlights the collaborative effort and individual responsibilities that collectively drove the project to fruition.

7.1 Data Management

Pranesh Jayasundar, Haniel Edward Jacob Thomson, and Jeya Varshini Bharath collaboratively handled data sourcing and preparation, playing a pivotal role in assembling and refining datasets for fine-tuning.

7.2 Model Development

Pranesh Jayasundar and Jeya Varshini Bharath jointly led the development and fine-tuning of the models, focusing on enhancing the

chatbot's proficiency in complex healthcare and emotional support tasks.

7.3 Testing and Quality Assurance

Haniel Edward Jacob Thomson and Jeya Varshini Bharath were responsible for testing and quality assurance, rigorously evaluating the chatbot's performance and ensuring its readiness for real-world deployment.

7.4 Documentation

Pranesh Jayasundar and Haniel Edward Jacob Thomson managed the comprehensive documentation of the project, including reporting stages and academic write-ups of the methodologies and results.

8 REFERENCES

(1) ChatCounselor: A Large Language Models for Mental Health Support.
https://arxiv.org/pdf/2309.15461.pdf
(2) Development and Evaluation of Three Chatbots for Postpartum Mood and Anxiety Disorders.
https://arxiv.org/pdf/2308.07407.pdf
(3) ChatDoctor: A Medical Chat Model Fine-Tuned on a Large Language Model Meta-AI (LLaMA) Using Medical Domain Knowledge
https://arxiv.org/abs/2303.14070
(4) Calibration of Transformer-based Models for Identifying Stress and Depression in Social Media
https://arxiv.org/abs/2305.16797