MediBot: Domain-Specific Fine-Tuning of Small Language Models for Clinical Triage

Presented by -

Hritik Hassani

The Challenge: Bottlenecks in Emergency Triage

Overwhelmed Emergency Rooms

Emergency departments face significant bottlenecks due to the volume of patients and the need for rapid, accurate assessment. This often leads to delays in care, increased patient anxiety, and potential for adverse outcomes.

Privacy & Cost Constraints

Leveraging large, general-purpose LLMs like GPT-4 for clinical applications presents critical hurdles. These include exorbitant API costs, data privacy concerns with patient information, and the inherent latency for real-time interactions.

The demand for quick, automated preliminary assessments in healthcare is growing, but existing solutions often fall short on accessibility, cost-efficiency, or data security.

Project Objective: Localized & Efficient Clinical Triage

Lightweight LLM for Triage

Our primary goal is to fine-tune a small language model, specifically TinyLlama-1.1B, to perform initial clinical triage assessments.

Consumer-Grade Hardware Deployment

The model is designed to operate efficiently on readily available consumer hardware, exemplified by a Google Colab T4 GPU, making it accessible and cost-effective.

Enhanced Privacy & Security

By enabling local deployment, MediBot significantly mitigates data privacy risks associated with transmitting sensitive patient information to external cloud-based services.

This project aims to demonstrate the viability of domain-specific fine-tuning for specialized medical applications, addressing key limitations of larger, more generic models.

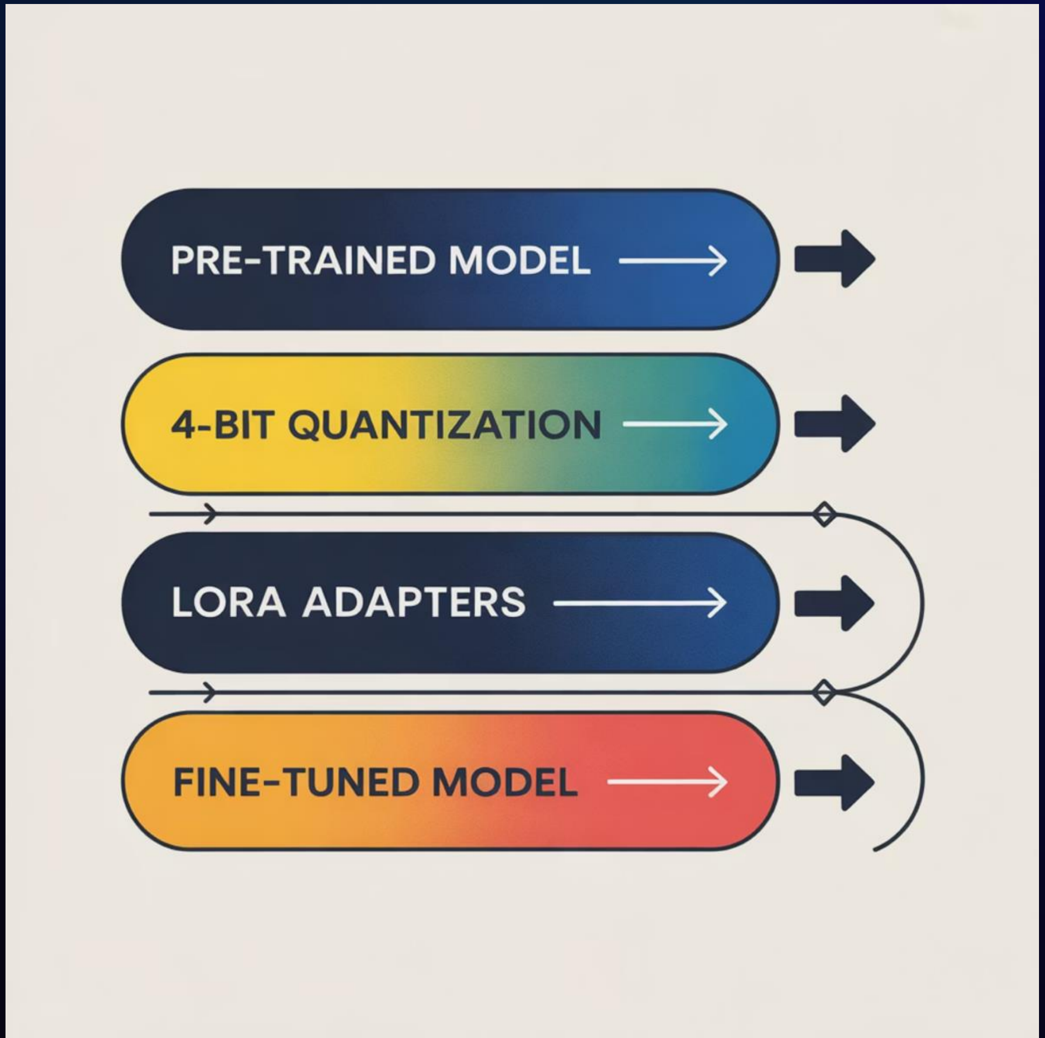
Methodology: Model Selection & Optimization

TinyLlama-1.1B: The Foundation

TinyLlama-1.1B was selected for its compact size and efficiency, striking a balance between computational demands and acceptable reasoning capabilities for basic clinical dialogue. While larger models offer superior general intelligence, TinyLlama's targeted fine-tuning allows it to excel within its domain without requiring extensive resources.

Resource-Efficient Fine-Tuning with QLoRA

To enable fine-tuning on a Google Colab T4 GPU (16GB VRAM), we employed QLoRA (Quantized Low-Rank Adaptation). This technique quantizes the model to 4-bit precision, drastically reducing memory footprint while preserving performance. QLoRA works by training only a small number of new parameters (adapters) that are then integrated with the pre-trained, quantized model.



QLoRA allows efficient fine-tuning of large models on consumer GPUs, making advanced AI more accessible.

Methodology: Domain-Specific Data Curation

The ruslanmv/ai-medical-chatbot Dataset

Our fine-tuning utilized the extensive `ruslanmv/ai-medical-chatbot` dataset, comprising approximately 250,000 high-quality question-answer pairs relevant to medical consultations and triage scenarios. This dataset covers a broad spectrum of symptoms, conditions, and general health inquiries, crucial for building a robust medical LLM.

Combating Hallucinations with System Prompts

A critical aspect of clinical AI is preventing 'hallucinations' or the generation of factually incorrect information. We implemented a rigorous 'System Prompt' format during training:

```
"You are a medical assistant chatbot.
Respond to the patient's symptoms and questions
with accurate medical information.
Do not provide diagnoses or prescribe medication.
Always advise consulting a qualified healthcare
professional.
Maintain a compassionate and professional tone."
```

This prompt was consistently prepended to training examples, guiding the model's behavior and ensuring responsible output.

Implementation: Training MediBot

Loading in 4-bit

The TinyLlama-1.1B model was loaded directly in 4-bit quantized format using the `bitsandbytes` library, a prerequisite for QLoRA and fitting within the T4 GPU's VRAM.

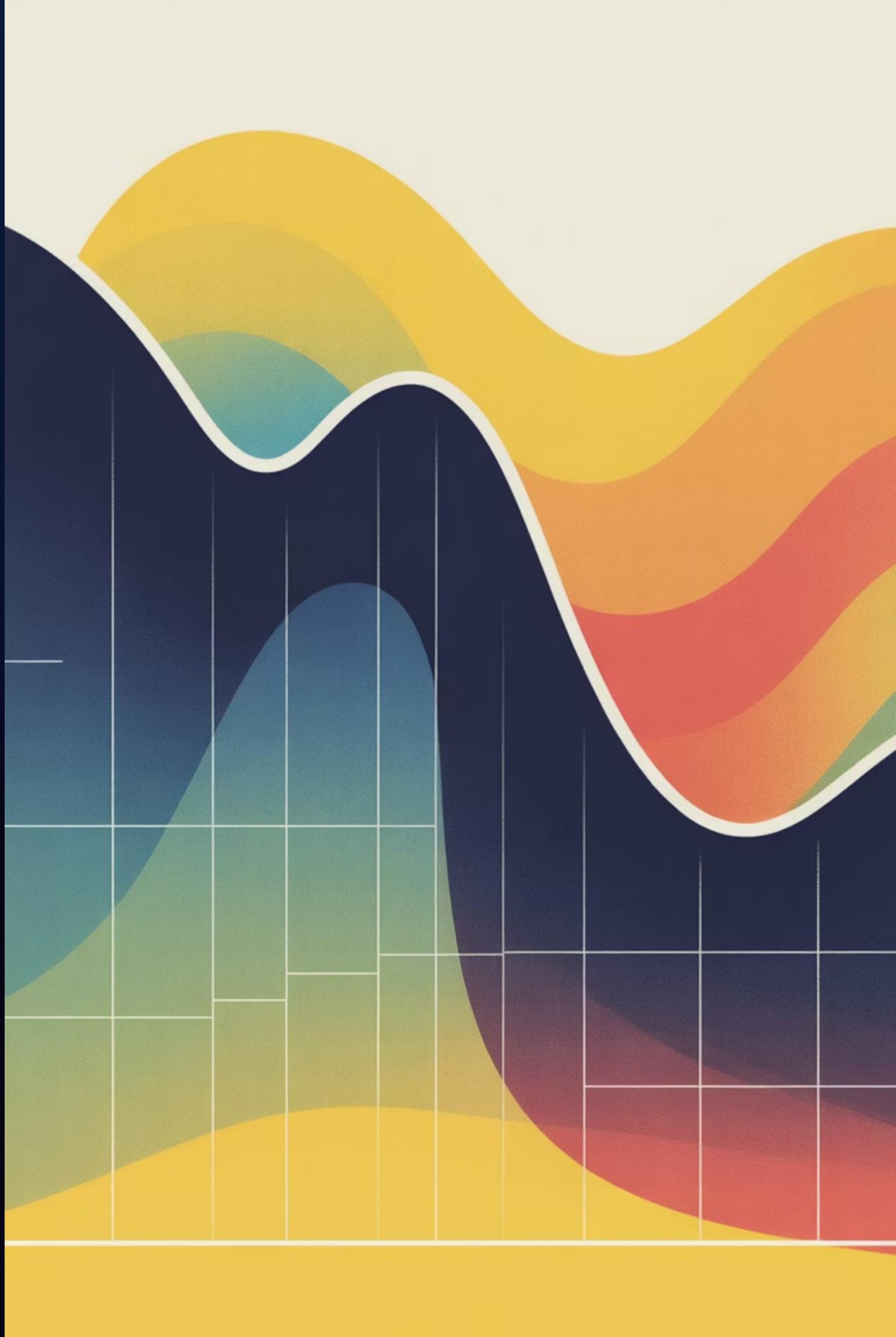
Applying LoRA Adapters

Low-Rank Adaptation (LoRA) adapters were applied with a rank of 16. This configuration balances fine-tuning effectiveness with a minimal number of trainable parameters, optimizing for performance on limited hardware.

Training Hyperparameters

- Learning Rate: 2e-4 (optimized for QLoRA)
- Batch Size: 4 (constrained by VRAM)
- Epochs: 3 (sufficient for domain adaptation)
- Optimizer: AdamW (standard for LLM training)

The training process was meticulously monitored to ensure convergence and prevent overfitting, leading to a stable and effective domain-specific model.



The Solution: Interactive Gradio Web UI

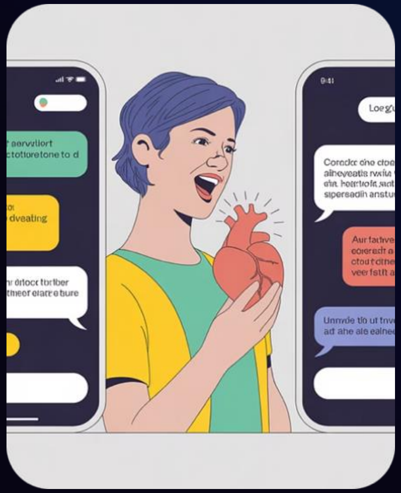
Intuitive User Interface

We developed a user-friendly web interface using Gradio, allowing seamless interaction with MediBot. The UI is designed to mimic a natural conversation flow, making it accessible for both medical professionals and patients for preliminary assessments.

- **Vitals Input:** Dedicated fields for essential patient vitals (e.g., age, gender, main complaint) to provide context for the LLM's responses.
- **Real-time Streaming Chat:** Responses from MediBot are streamed in real-time, enhancing the interactive experience and reducing perceived latency.
- **Critical Disclaimer:** A prominent disclaimer reiterates that MediBot is an assistant, not a diagnostic tool, and professional medical advice should always be sought.

The Gradio interface ensures accessibility and responsible use of the AI assistant in a clinical context.

Results: Qualitative Assessment & Performance



Scenario 1: High Urgency (Heart Attack)

MediBot accurately identified symptoms indicative of a potential heart attack, immediately advising the user to seek emergency medical attention and outlining critical first steps. The response demonstrated appropriate urgency and caution.

Scenario 2: Low Urgency (Scraped Knee)

For a common minor injury like a scraped knee, MediBot provided sensible advice on cleaning the wound, applying antiseptic, and recognizing signs of infection, without escalating unnecessarily.

Inference Speed: Speed: 35 Tokens/Sec

Running on the Google Colab T4 GPU, MediBot consistently achieved an inference speed of approximately 35 tokens per second. This rate provides a near real-time conversational experience, crucial for responsive triage applications.

Qualitative analysis across various scenarios confirmed MediBot's ability to provide contextually appropriate and responsible medical guidance, adhering to the system prompt's constraints.

Challenges & Ethical Considerations

Hallucinations

Initial models often generated plausible but incorrect information. This was largely mitigated by the strict 'System Prompt' engineering during training, guiding the model's factual accuracy and adherence to its role.

Bias in Data

The dataset's inherent biases (e.g., demographic representation, symptom presentation) could lead to skewed responses. Ongoing efforts involve diversifying training data and implementing fairness metrics.

Automation Bias

Over-reliance on AI outputs can lead to human clinicians overlooking critical details. The UI's prominent disclaimer and the assistant-only role are designed to counteract this bias.

Data Privacy

Even with local deployment, robust security measures are paramount. Ensuring patient data remains secure during input and processing is a continuous focus, leveraging encrypted communication and local storage solutions.

Addressing these challenges is vital for the responsible deployment of AI in healthcare, emphasizing that MediBot is a tool t o augment, not replace, human expertise.

Conclusion & Future Directions

1 Successful Proof of Concept

MediBot demonstrates the feasibility of fine-tuning small language models for domain-specific clinical triage on accessible consumer hardware, running effectively within Google Colab's free tier.

2 Enhanced Accuracy & Efficiency

The domain-specific fine-tuning, combined with a robust system prompt, significantly improves the accuracy and safety of responses, making MediBot a valuable preliminary assessment tool.

3 Future Work: Retrieval-Augmented Generation (RAG)

Integrating RAG will allow MediBot to query external, up-to-date medical databases, providing more current and evidence-based information, thereby reducing reliance on pre-trained knowledge.

4 Future Work: Voice Support & Multimodality Multimodality

Adding voice input/output capabilities would enhance accessibility and user experience, while exploring multimodal inputs (e.g., image analysis for dermatological conditions) could expand MediBot's utility.

MediBot represents a step towards democratizing access to AI-powered medical assistance, prioritizing privacy, cost-effectiveness, and responsible AI deployment.