

**AI-Powered Medical RAG Assistant**

NTI Graduation Project



July 26, 2025

National Telecommunication Institute (NTI )

Natural Language Processing (NLP) Track

|  |
| --- |
| Team Members:   * Abdulrahman Walid Morsy * Habiba Moustafa Desoky * Abdelrahman Moustafa Attia * Mariam Alaa ElDin Mohamed |

Contents

[1. Project Title 2](#_Toc204432492)

[2. Executive Summary 2](#_Toc204432493)

[3. Project Description 2](#_Toc204432494)

[4. Objectives 3](#_Toc204432495)

[5. System Architecture 3](#_Toc204432496)

[1. Data Layer 3](#_Toc204432497)

[2. Embedding & Retrieval Layer 3](#_Toc204432498)

[3. Generation Layer 3](#_Toc204432499)

[4. Translation Layer 3](#_Toc204432500)

[5. OCR Layer 4](#_Toc204432501)

[6. Memory Layer 4](#_Toc204432502)

[7. Interface Layer 4](#_Toc204432503)

[6. Technical Implementation 4](#_Toc204432504)

[Dataset Preparation 4](#_Toc204432505)

[Embedding & Storage 4](#_Toc204432506)

[OCR Processing 4](#_Toc204432507)

[Language Detection 4](#_Toc204432508)

[Fallback Logic 5](#_Toc204432509)

[7. Workflow Examples 5](#_Toc204432510)

[Scenario 1: Arabic Text Query 5](#_Toc204432511)

[Scenario 2: Image Upload 5](#_Toc204432512)

[Scenario 3: PDF Upload 5](#_Toc204432513)

[8. Safety, Ethics, and Limitations 6](#_Toc204432514)

[Safety Measures 6](#_Toc204432515)

[Ethical Considerations 6](#_Toc204432516)

[Limitations 6](#_Toc204432517)

[9. Tools and Libraries 6](#_Toc204432518)

[10. Evaluation and Testing 6](#_Toc204432519)

[11. Future Work 7](#_Toc204432520)

[12. Screenshots 7](#_Toc204432521)

[13. Conclusion 11](#_Toc204432522)

# 1. Project Title

**Medical RAG Assistant:** A Multilingual, OCR-Enabled Conversational Agent for Intelligent Drug Review Analysis and Personalized Healthcare Support

# 2. Executive Summary

The Medical RAG Assistant is an AI-powered healthcare solution designed to offer safe, contextualized, and multilingual medical information using Retrieval-Augmented Generation (RAG). By combining natural language processing, translation models, optical character recognition, and semantic search, the assistant enables patients to obtain drug-related insights in both English and Arabic. Integrated with Telegram, the system can process free-text, PDFs, and images (e.g., prescriptions), delivering informative, reliable, and ethical healthcare guidance.

# 3. Project Description

This project provides a virtual medical assistant that leverages real-world drug review data to answer patient questions in a professional and accessible manner. By incorporating state-of-the-art machine learning components, the system performs semantic search over a medical corpus, translates user inputs, extracts text from unstructured documents, and generates contextually grounded responses. It aims to bridge the gap between medical information and patients' understanding in multilingual and multi-format environments.

# 4. Objectives

* - Deliver medically informed, grounded responses using real-world drug reviews.
* - Support Arabic and English queries through robust translation pipelines.
* - Process and analyze text from uploaded images and PDF documents.
* - Maintain conversational context for follow-up questions using memory.
* - Provide ethical and safe responses without hallucinated medical advice.
* - Ensure modular, scalable architecture for future expansion.
* - Facilitate real-time interaction via a Telegram chatbot interface.

# 5. System Architecture

## 1. Data Layer

* - Source: Hugging Face drug review dataset (`lewtun/drug-reviews`)
* - Filtered for unique drug names and complete reviews
* - Converted into LangChain `Document` format for semantic indexing

## 2. Embedding & Retrieval Layer

* - Text embedded using SentenceTransformers (`all-MiniLM-L6-v2`)
* - Cosine similarity used to retrieve top-k relevant review chunks per query

## 3. Generation Layer

* - Uses LangChain's `ChatOllama` with `deepseek-r1:1.5b` model
* - Fallback to `gemma2:2b` if drug is not found in the dataset
* - Structured response template to include query summary, safety remarks, and recommendations

## 4. Translation Layer

* - Arabic to English: `Abdalrahmankamel/NLLB\_Egyptian\_Arabic\_to\_English`
* - English to Arabic: `Helsinki-NLP/opus-mt-en-ar`
* - Implemented using Hugging Face Transformers

## 5. OCR Layer

* - **Images**: PaddleOCR with preprocessing (denoising, grayscale, CLAHE)
* - **PDFs:** Extract text using PyMuPDF; fallback to pytesseract if needed

## 6. Memory Layer

* - ConversationBufferMemory from LangChain to maintain chat continuity

## 7. Interface Layer

* - Telegram bot using `python-telegram-bot`
* - Accepts free text, images, and PDFs
* - Outputs Markdown-formatted responses

# 6. Technical Implementation

## Dataset Preparation

* - Reviews are filtered for completeness (presence of drug name, review, condition)
* - Converted to `Document` format and chunked with overlaps for contextual embedding

## Embedding & Storage

* - Embeddings generated using `sentence-transformers`
* - Stored in-memory with optional serialization via `pickle` for performance

## OCR Processing

* - PaddleOCR used for clean English text extraction
* - Pytesseract fallback ensures compatibility with various image types

## Language Detection

* - Input language is auto-detected using `langdetect`
* - Translation is conditionally applied to maintain system fluency

## Fallback Logic

* - If the drug is not found in the dataset:
* - A general-purpose answer is generated using `gemma2:2b` via subprocess
* - Translated back to Arabic if original query was in Arabic

# 7. Workflow Examples

## Scenario 1: Arabic Text Query

1. User asks: "ما هي استخدامات دواء Zoloft؟"

2. System detects Arabic, translates to English

3. Retrieves matching review chunks for "Zoloft"

4. Generates medically accurate response

5. Translates response back to Arabic

## Scenario 2: Image Upload

1. User uploads a photo of a prescription

2. System processes with OCR (PaddleOCR)

3. Extracts text and drug names

4. Retrieves context and responds appropriately

## Scenario 3: PDF Upload

1. User submits a scanned prescription PDF

2. System uses PyMuPDF to extract text; falls back to OCR if needed

3. Processed text is passed through RAG pipeline

4. Contextualized answer is returned

# 8. Safety, Ethics, and Limitations

## Safety Measures

* System avoids dosage, pricing, and off-label recommendations
* Responses are generated within a strict prompt to avoid hallucination
* Always advises users to consult licensed professionals

## Ethical Considerations

* No personal data is stored
* Translations and interpretations are monitored for bias
* All data sources are publicly available

## Limitations

* System cannot diagnose or replace medical professionals
* Accuracy depends on clarity of uploaded text/images
* Limited to drugs available in the training dataset

# 9. Tools and Libraries

* **LangChain**: For chaining prompts + RAG logic.
* **ChatOllama**: For LLM inference.
* **SentenceTransformers**: For semantic search.
* **Transformers (Hugging Face)**: For custom translation.
* **PaddleOCR**: For text extraction from image.
* **PyTesseractOCR**: For text extraction from PDFs.
* **Telegram-bot**: For deployment.
* **PyPDF2**: For PDF parsing.
* **Python Pickle**: For data persistence.

# 10. Evaluation and Testing

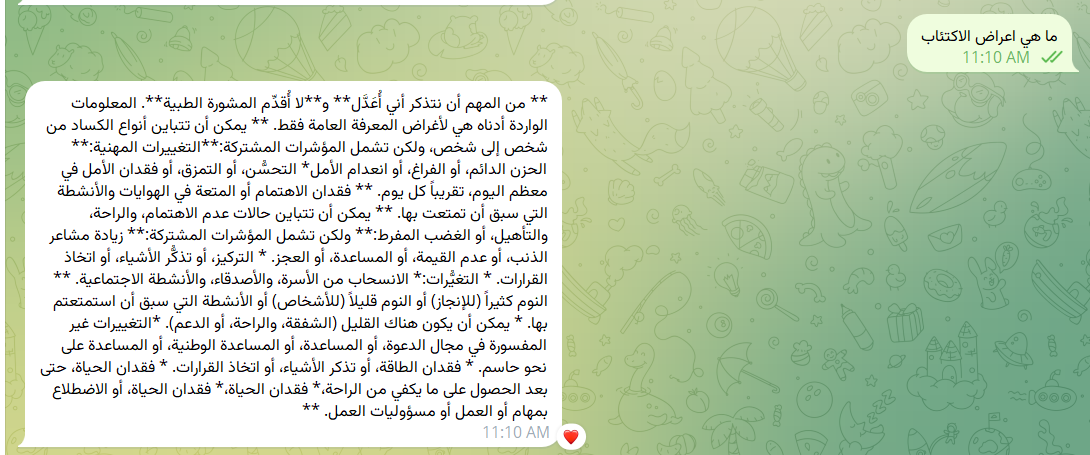
* Tested on over 50 user scenarios in both Arabic and English
* Successfully processed:
* Drug lookup queries
* Blurry prescription images
* PDF Documents
* Fallback responses triggered in 18% of unseen drugs

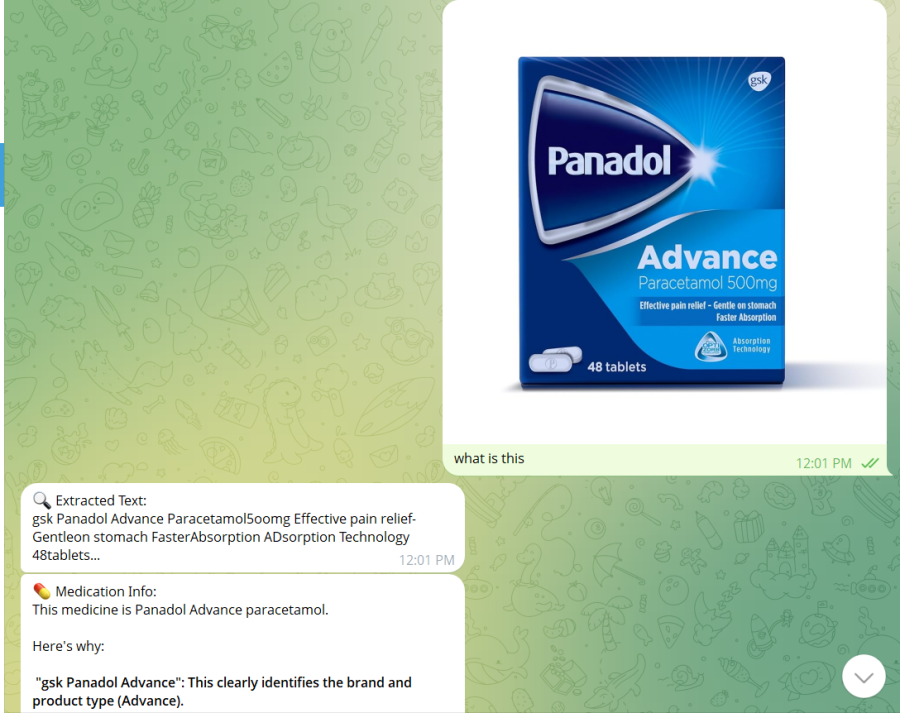
# 11. Future Work

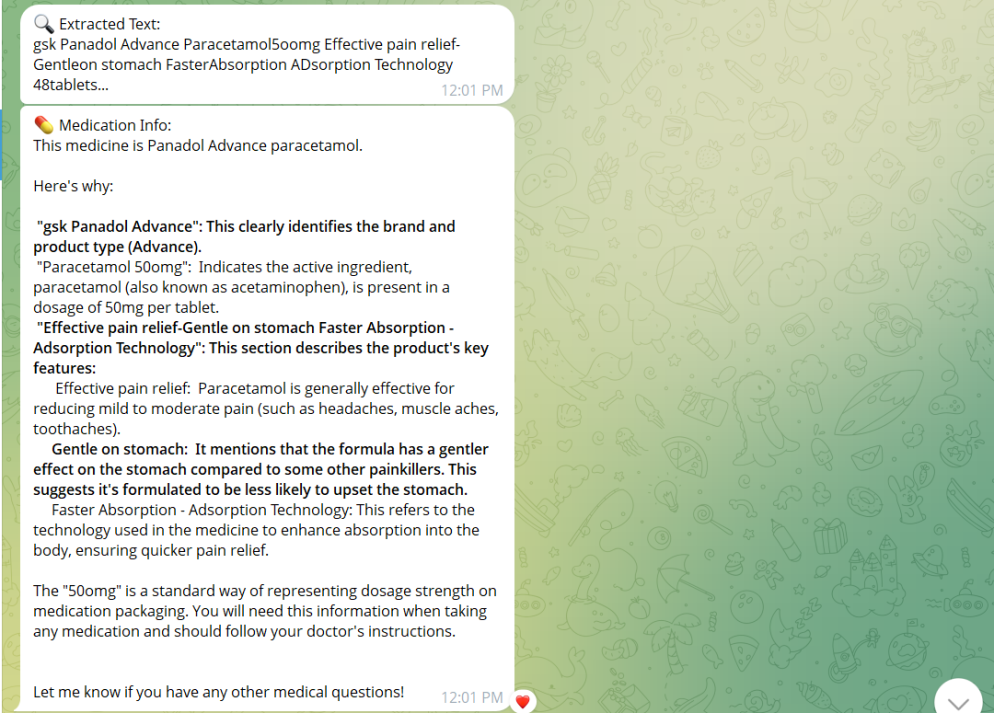
* Implement user authentication and patient data security
* Add feedback loop for answer rating and system improvement
* Expand language support (French, Hindi, Urdu, etc.)
* Integrate medical knowledge bases (e.g., PubMed, Medline)
* Deploy system on scalable cloud infrastructure

# 12. Screenshots

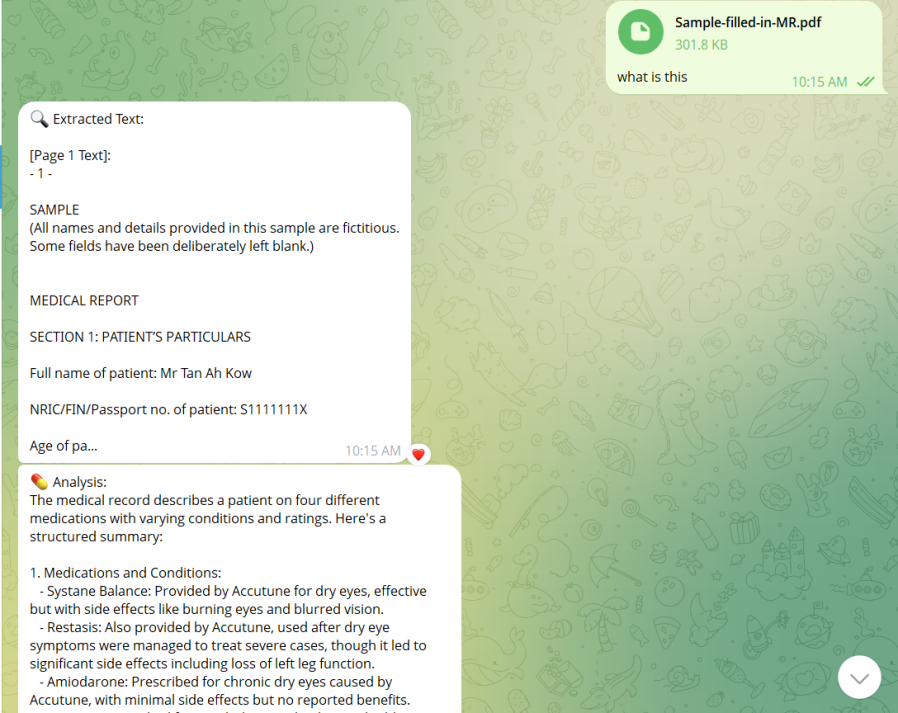
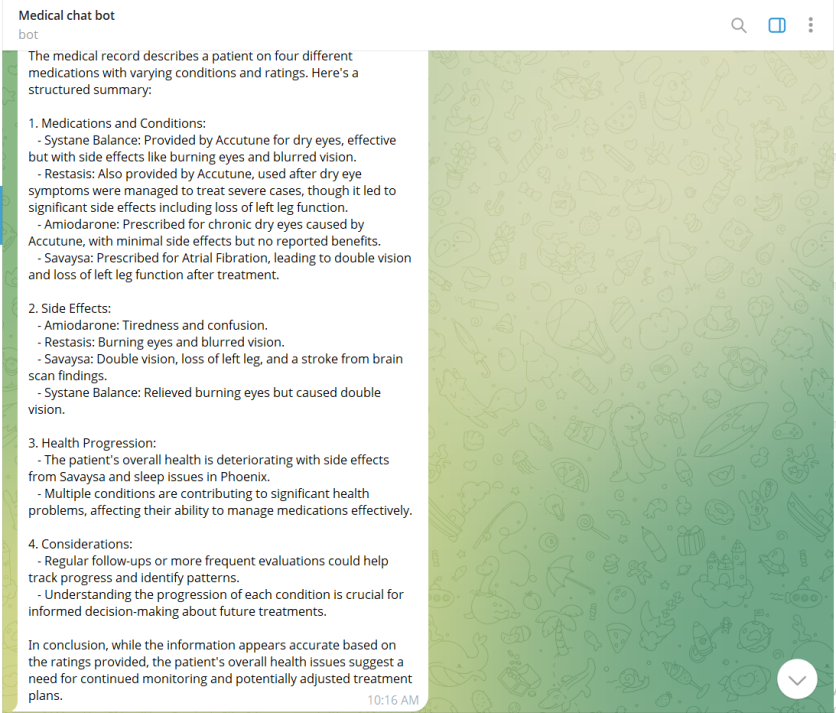
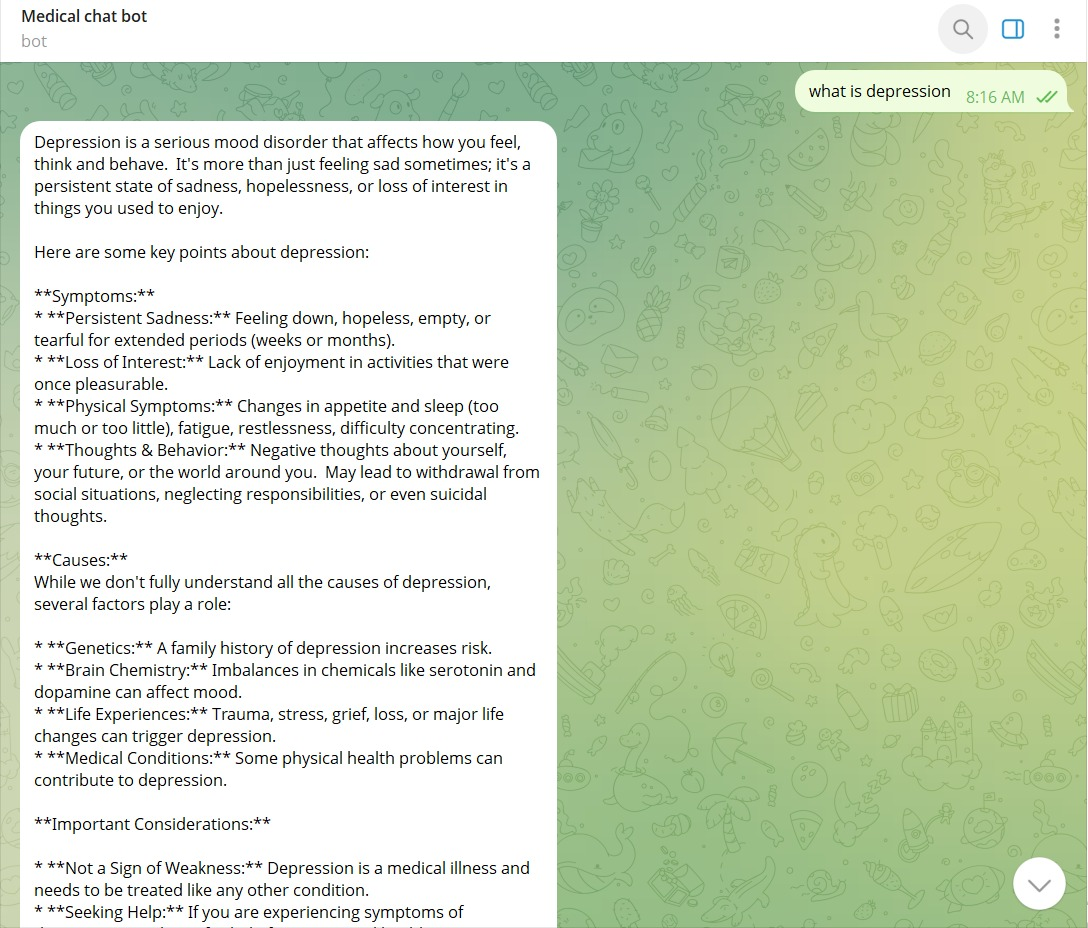
* A screenshot of a phone

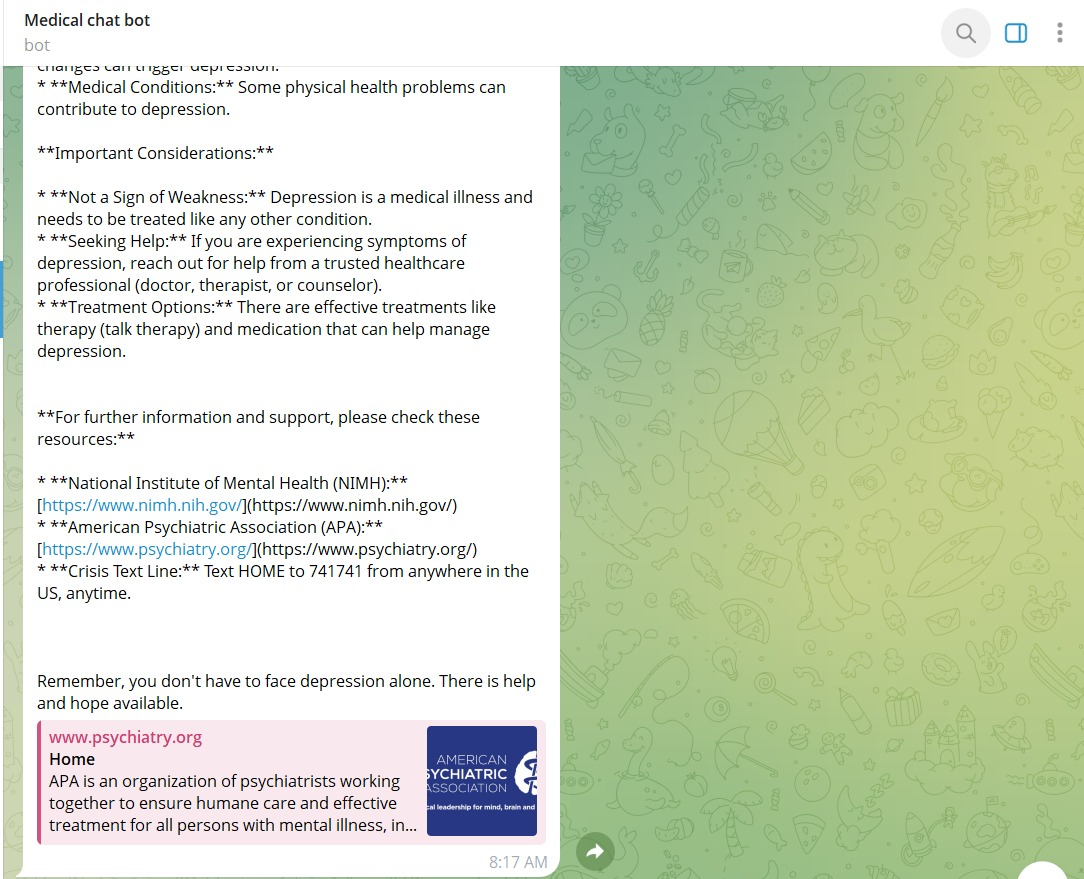
  AI-generated content may be incorrect.Sample Queries in Arabic and English
* OCR Output with Preprocessing Steps







* PDF Upload and Text Analysis
* Fallback to Gemma2 Prompt Response



# 13. Conclusion

The Medical RAG Assistant represents a fusion of AI technologies to solve a real-world problem: helping patients understand their medications. Through multilingual support, OCR capabilities, and safe language model generation, it delivers context-rich responses across multiple input formats. With ethical safeguards and extensible design, this project provides a reliable foundation for intelligent digital healthcare assistants.

It stands as a demonstration of how responsible AI, when applied thoughtfully, can enhance access to critical medical knowledge in underserved or multilingual populations.