Machine Learning Retrieval-Augmented Generation System for Medical Analysis

SWE402 - Senior Design Project

Yiğit SERT
B201202041

Sakarya University
Faculty of Computer and Information Sciences
Software Engineering

Prof. Dr. Devrim AKGÜN

Introduction

Today’s healthcare systems collect vast amounts of data from devices, hospitals, and surveys, but turning this information into personalized and easy-to-understand advice remains a significant challenge. Most health advice is either too general to be useful, while advanced AI tools are often too technical for the average person.

This presentation explains a system designed to solve this problem by combining the clear logic of decision trees with the power of Large Language Models (LLMs) to create advice that is both accurate and simple for everyday users. To explain how this is achieved, we’ll cover five main parts: the problem, related projects, how the proposed system works, results from experiments, and final thoughts with future ideas.

Why LLMs & Decision Trees?

Large Language Models (LLMs) are smart AI tools that can read and write like a human. In healthcare, they help with:

- Writing clinical notes
- Answering health questions
- Talking to patients in simple language

They are good at making technical health rules sound natural and helpful.

Decision trees explain their reasoning with clear if-then rules. But these rules can still be too technical for users. That's why we bring in LLMs—to turn these rules into friendly, understandable advice that people can actually use. This mix helps build trust and makes the technology more useful.

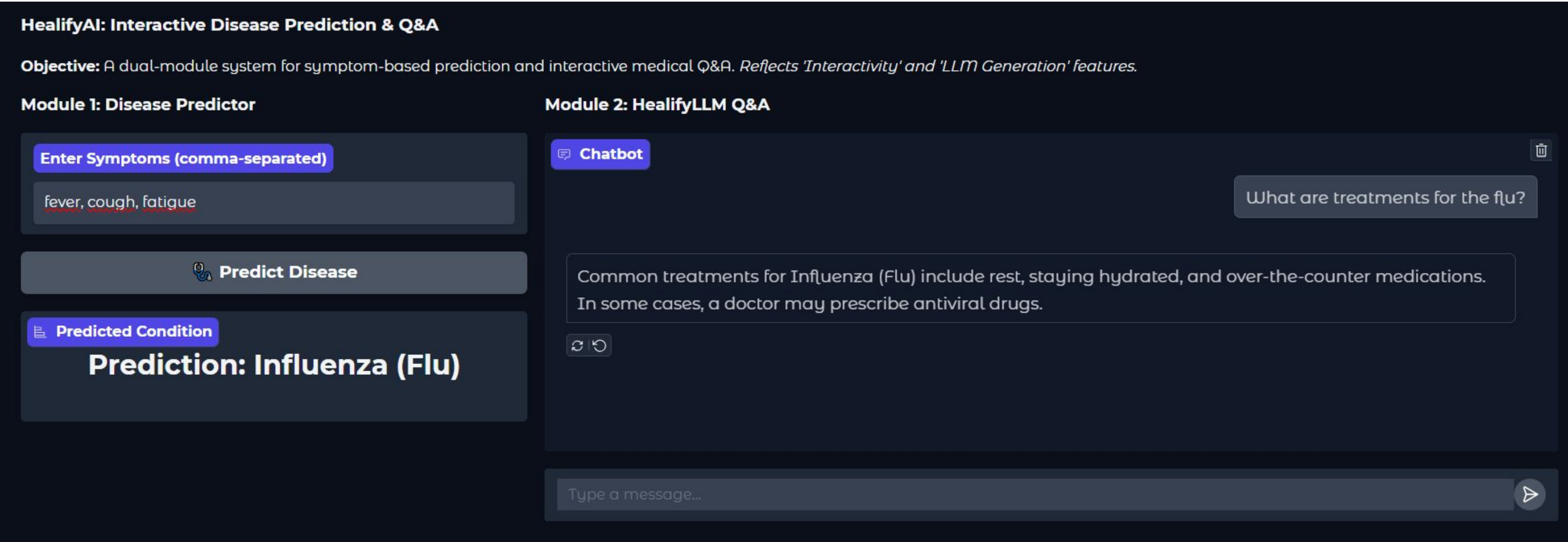
Case Study – HealifyAI

HealifyAI is a health system that predicts diseases and answers user questions. It uses:

- A Random Forest model to predict diseases
- A RoBERTa model to give answers in plain language
- A simple web interface (Gradio)

It’s focused on being accurate and understandable.

Case Study – HealifyAI Mock-Up



Case Study – MLRAG

MLRAG shows how decision tree rules and LLMs can work together. It uses a system called RAG (Retrieval-Augmented Generation). It combines:

- A health dataset about heart disease
- Decision rules from a tree model
- Gemini LLM to write health advice

This approach helps create advice tailored to each person’s health data.

Case Study – MLRAG Mock-Up

AI Medical Researcher (MLRAG)

Objective: Use extracted decision rules and a patient profile to generate personalized recommendations. *Reflects 'Rule Extraction' and 'LLM Generation' features.*

Input Data (Pre-defined)

Kod

```
Patient Profile:
{
    "Age": 67, "Sex": "Male", "BMI": 31.5,
    "Smoking": "Yes", "SleepTime": 5
}

Extracted Rules (from Decision Tree):
1. IF BMI > 30 AND Smoking == 'Yes' -> High risk of HeartDisease.
2. IF SleepTime <= 5 -> Increased risk of Stroke.
```

Generate Personalized Recommendation

LLM-Generated Recommendation

--- Personalized Health Analysis for Patient ---

This analysis is based on your profile (Age: 67, BMI: 31.5, Smoker: Yes) and established risk models.

Key Findings from Your Profile:

1. Your BMI of **31.5** is in the obese category. Combined with your smoking habit, this places you at a significant risk for heart disease (as per Rule 1).
2. Your sleep duration of **5 hours** is below the recommended 7-9 hours, a known risk factor for cardiovascular events (as per Rule 2).

Actionable Recommendations:

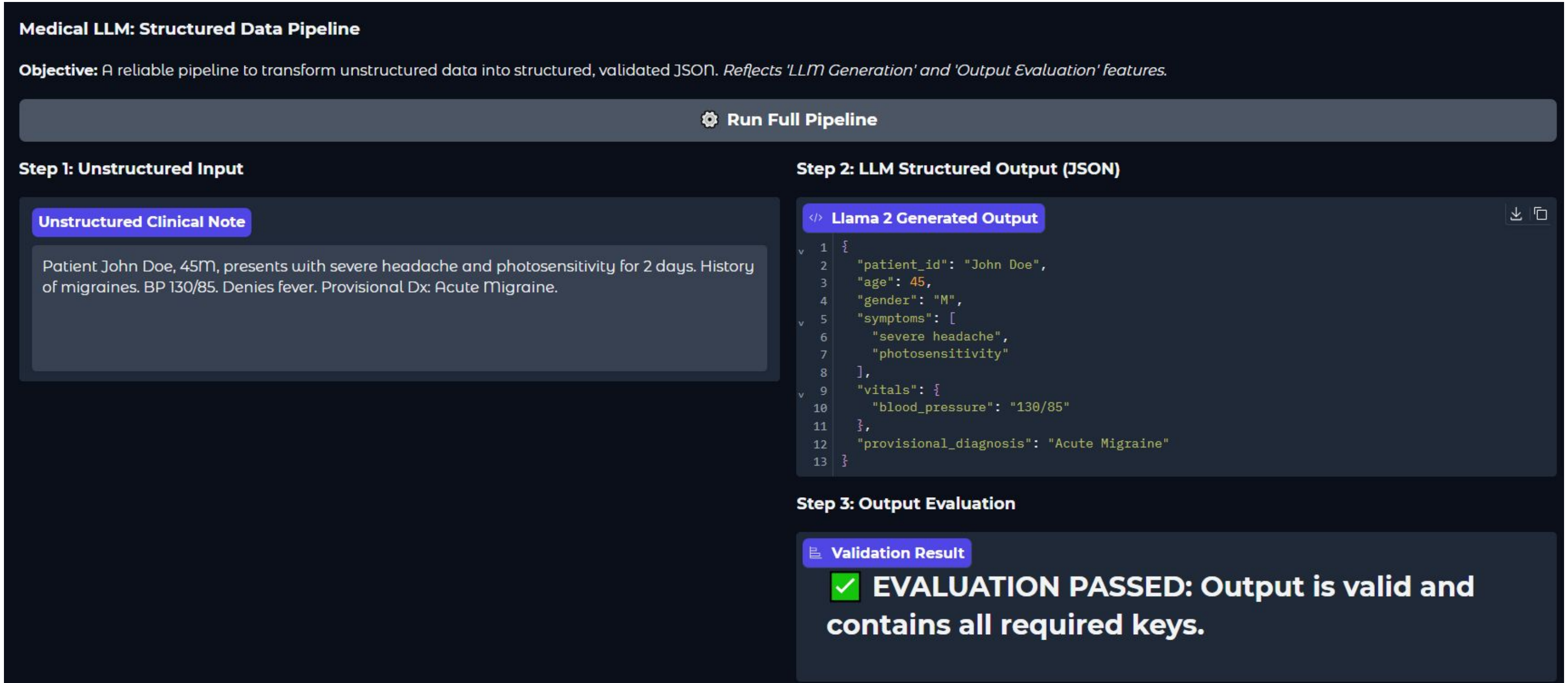
- **Smoking Cessation:** This is the most critical step. We recommend consulting your doctor about cessation programs.
- **Diet and BMI Management:** Focus on a balanced diet to gradually lower your BMI.
- **Improve Sleep Hygiene:** Aim for at least 7 hours of sleep per night.

Case Study – Medical LLM

This system uses the Llama 2 model to read different types of health data (like tables, reports, and scan details) and turn them into organized formats like JSON. It focuses on:

- Accuracy
- Privacy
- Easy-to-use outputs for health professionals

Case Study – Medical LLM Mock-Up

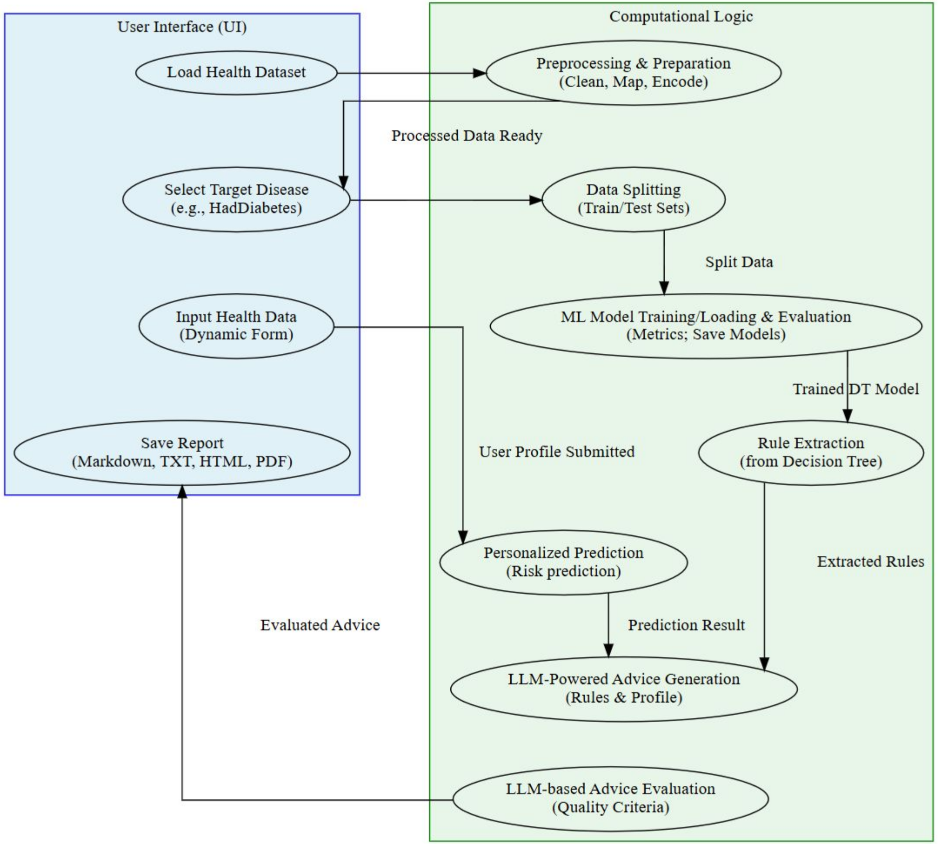


Case Study Comparison Table

Feature	HealifyAI	MLRAG	Medical LLM	Thesis Model
Rule Extraction	✖	✔	✔	✔
LLM Generation	✔	✔	✔	✔
Interactivity	✔	✖	✖	✔
Output Evaluation	✖	✖	✔	✔

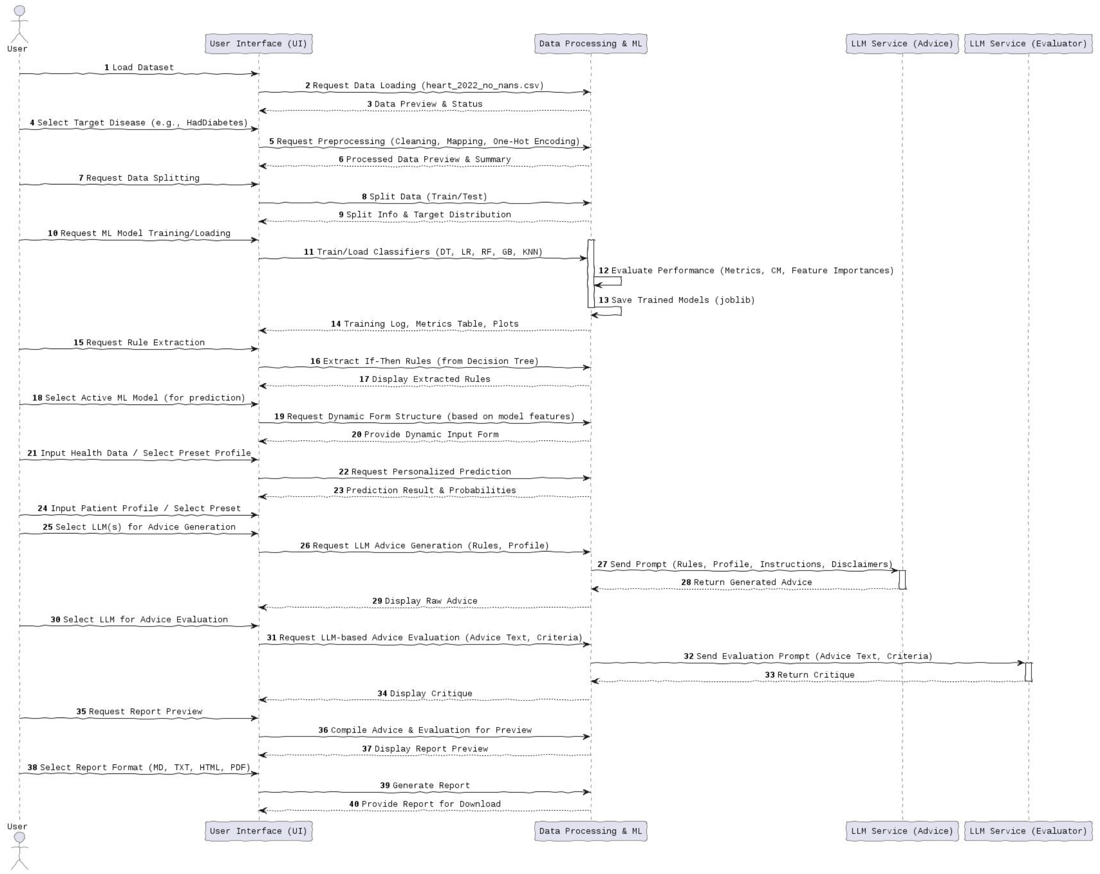
System Architecture

The system follows a clear data pipeline starting from data loading and preprocessing (using a heart disease dataset), continuing through feature engineering, encoding, and model training. Stratified splitting ensures balanced data, and the training process involves several well-known models such as Decision Trees, Logistic Regression, Random Forest, Gradient Boosting, and K-Nearest Neighbors.



System Workflow

This sequence diagram outlines a machine learning system where a User interacts with a UI to manage data and models. The Data Processing & ML component handles dataset loading, preprocessing, model training, and evaluation. Users can extract rules from models, select an active model for personalized predictions based on health data, and leverage LLM Services for advice generation and evaluation. The process culminates in the generation of detailed reports.

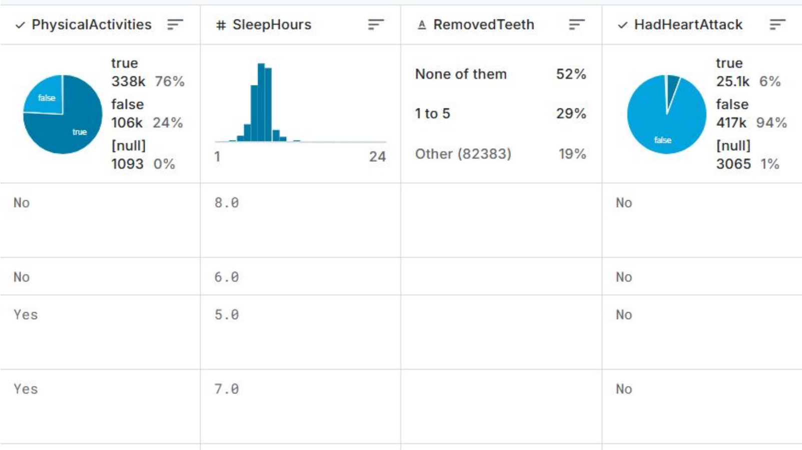


Dataset & Preprocessing

The system uses a dataset from a public health survey (BRFSS 2022). It includes things like BMI, diabetes, and lifestyle habits. The steps taken include:

- Renaming messy columns
- Turning categories into numbers
- Picking the disease to predict
- One-hot encoding to prepare features

This makes the data ready for training the models.



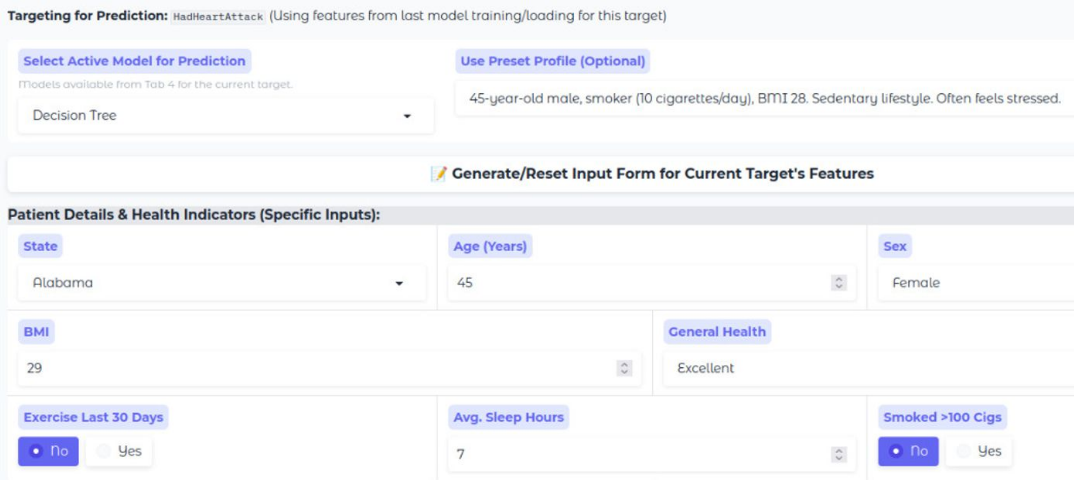
ML Models and Metrics

The system trains and evaluates a suite of machine learning models on the preprocessed and split dataset. The user selects the target disease in a prior step, and models are trained specifically for that target.

Models	Evaluation Metrics
Decision Tree	Accuracy
Logistic Regression	Precision
Random Forest	Recall
Gradient Boosting	F1-Score
K-Nearest Neighbors	Confusion Matrix

ML Prediction

Users can select from trained ML models to make a detection prediction for the chosen disease. They can either pick a preset patient profile or enter custom parameters manually. This allows dynamic, personalized prediction requests before generating advice.



Interpretable Rule Extraction

Beyond predictive performance metrics, another objective of this study was to ensure model interpretability, particularly for the Decision Tree classifier, whose outputs directly inform the LLM-based advice generation. Following the training and evaluation of the Decision Tree model for a selected target (e.g., ‘HadHeartAttack‘), if-then rules were extracted. These rules represent the logical pathways the model learned from the data to arrive at a prediction.

```
— ChestScan_Yes <= 0.50
  — DifficultyWalking_Yes <= 0.50
    — AgeCategory_Age 80 or older <= 0.50
      — GeneralHealth_Fair <= 0.50
      — class:  0 // Indicates lower risk
      — GeneralHealth_Fair > 0.50
      — class:  1 // Indicates higher risk
```

Prompt Template Design

To guide the LLM, a structured prompt is used that includes:

- A short task description
- The patient's profile (age, habits, etc.)
- The decision rules
- Instructions (don't give diagnoses, suggest seeing a doctor)

This setup ensures the advice is friendly, safe, and personalized.

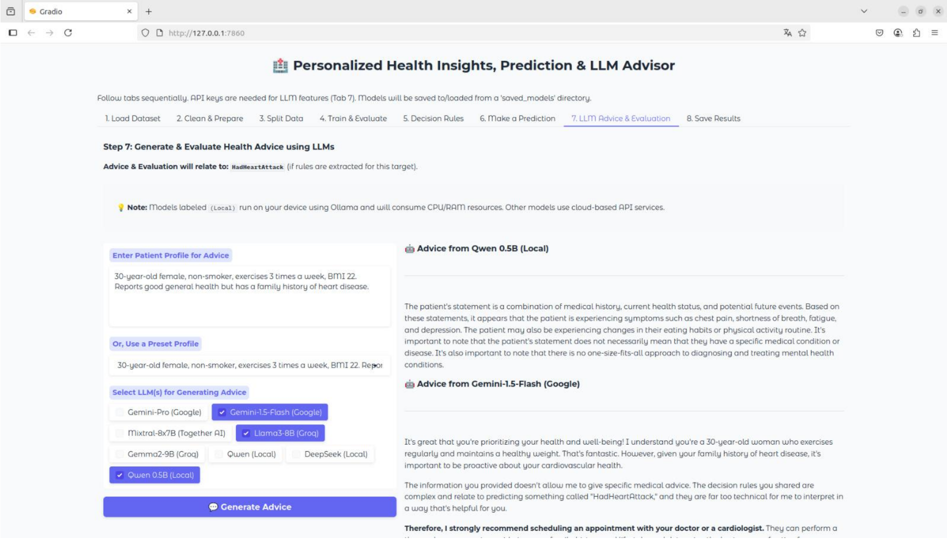
```
def generate_advice_for_models_tab(profile, selected_llms, progress=gr.Progress(track_tqdm=True))
    global generated_advice_text, advice_evaluation_text, rules, current_target_disease, MODEL_CH
    if not profile or not selected_llms:
        return gr.update(value="Please provide a patient profile and select at least one LLM."),

    advice_evaluation_text = ""
    advice_outputs = []

    # Construct a more detailed prompt
    rules_context = f"For context, here are some decision rules related to predicting '{current_t
    prompt = f"Based on the following patient profile and decision-rule context, provide clear, e
```

```
rently available for context."
onsulting a doctor.\n\n**Patient Profile:**\n{profile}\n\n**Contextual Rules:**\n{rules_context}"
```

LLM Integration

The system connects with different LLMs.

Users can pick which LLM to use. The system sends a prompt and shows the answer.



Evaluation of LLM Outputs

Flesch-Kincaid Scores are key tools to evaluate how easily English text can be understood—crucial for safe and effective health communication. Complex language may cause misinterpretation, especially for those with low health literacy. Additionally, the time it takes for each model to respond was also recorded to compare performance under real-time conditions.

Flesch Reading Ease:
Rates text on a 0–100 scale.
60–70 = plain English, suitable for 8th–9th grade level.
Below 60 = possibly too complex.

Flesch-Kincaid Grade Level:
Converts readability into a U.S. school grade level.
Target: Grade 8–9 for public health materials to ensure wide accessibility.

Evaluation of LLM Outputs

Gemma2-9B and Mixtral-8x7B consistently delivered more readable advice, achieving higher Flesch Reading Ease scores (50–59). Gemini-1.5-Flash produced slightly denser text, while Llama3-8B's outputs were the least readable, often requiring a higher reading level (FK Grade >12). These results suggest a trade-off between linguistic simplicity and model complexity, with Groq models favoring speed and Gemini favoring completeness.

Scenario	LLM Advice Generator	Time (s)	Flesch Ease	FK Grade
General Wellness Seeker	Mixtral-8x7B (Together AI)	5.09	59.7	8.2
General Wellness Seeker	Llama3-8B (Groq)	1.02	42.1	12.4
General Wellness Seeker	Gemini-1.5-Flash (Google)	5.07	48.9	9.9
General Wellness Seeker	Gemma2-9B (Groq)	0.76	57.3	8.8
Diabetic with Heart Concerns	Mixtral-8x7B (Together AI)	8.71	46.9	10.8
Diabetic with Heart Concerns	Llama3-8B (Groq)	1.21	47.6	11.3
Diabetic with Heart Concerns	Gemini-1.5-Flash (Google)	5.22	43.2	11.1
Diabetic with Heart Concerns	Gemma2-9B (Groq)	1.01	55.4	9.7
Young Adult w/ Family History	Mixtral-8x7B (Together AI)	1.86	49.4	11.5
Young Adult w/ Family History	Llama3-8B (Groq)	1.21	33.0	14.9
Young Adult w/ Family History	Gemini-1.5-Flash (Google)	4.54	43.7	10.7
Young Adult w/ Family History	Gemma2-9B (Groq)	1.01	50.5	9.9
Elderly with Arthritis	Mixtral-8x7B (Together AI)	8.59	41.8	11.4
Elderly with Arthritis	Llama3-8B (Groq)	1.56	29.2	14.7
Elderly with Arthritis	Gemini-1.5-Flash (Google)	5.01	44.3	10.6
Elderly with Arthritis	Gemma2-9B (Groq)	1.09	50.0	9.5

Note: Time (s) refers to Advice Generation Time. FK Grade is Flesch-Kincaid Grade Level.

Evaluation of LLM Outputs - Judge

A structured evaluation by *Gemini-1.5-Flash* (as the LLM Judge) rated four advice-generating LLMs on key criteria from 1 to 5. The assessment covered clarity, relevance, safety, completeness, tone, and more.

Criterion	**Mixtral-8x7B (Together AI)**	**Llama3-8B (Groq)**	**Gemini-1.5-Flash (Google Custom)**	**Gemma2-9B (Groq)**
Clarity	4.6	4.1	**4.9**	4.4
Actionability	4.4	4.0	**4.7**	3.9
Safety	**5.0**	**5.0**	**5.0**	**5.0**
Relevance	4.7	4.4	**4.9**	4.6
Completeness KI	4.6	4.1	**4.9**	4.0
No Diagnosis	**5.0**	**5.0**	**5.0**	**5.0**
Encourage Consult	**5.0**	**5.0**	**5.0**	**5.0**
Empathy Tone	**4.5**	3.9	4.4	4.1
Overall Average	**4.60**	4.30	**4.84**	4.38

Note: Scores are illustrative averages derived from the LLM Judge's structured JSON outputs. KI: Key Information checklist coverage.

Future Directions

Future work will prioritize making the system more robust and practical. This involves enhancing evaluation with multi-assessor studies, testing a wider range of LLMs and user scenarios, and optimizing the RAG process. Crucially, the focus will shift to user-centric studies to assess real-world utility and to bolstering system resilience with better error handling to ensure reliable operation. Also the results should be evaluated, labeled and confirmed by authorized specialists.

THANK YOU