# Healthcare Correspondence LLM

CS 4485 / Spring 2024
Department of Computer Science
Erik Jonsson School of Engineering & Computer Science
The University of Texas at Dallas
Richardson, TX 75080, USA

Hyeongjin Joo
hyeongjin.joo@utdallas.edu

Miguel Gebremedhin
miguel.gebremedhin@utdallas.edu

Jonathan Asplund
jonathan.asplund@utdallas.edu

Faculty Advisor
Dr. Yi Zhao

Ari Hu
ari.hu@utdallas.edu

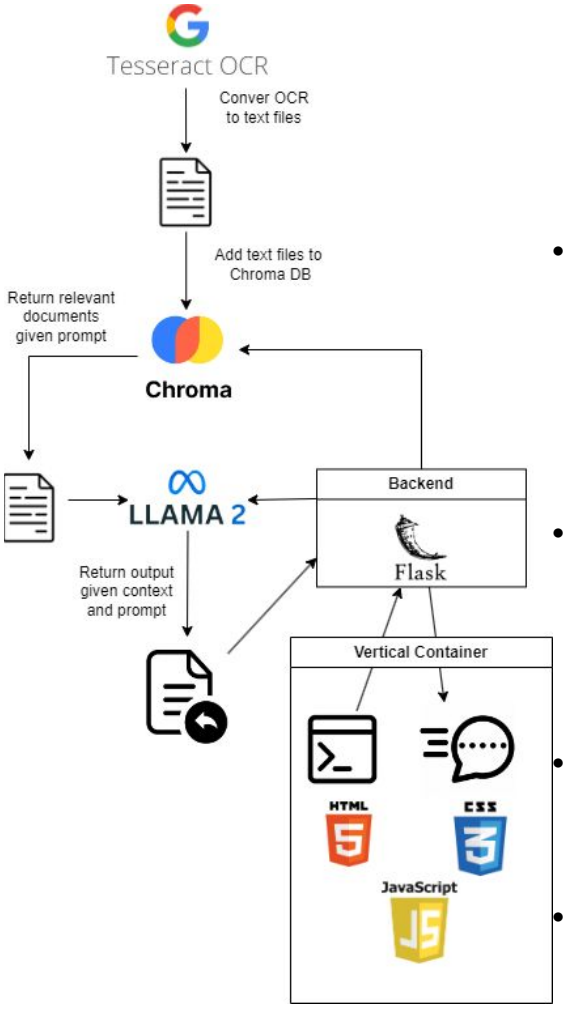
Ethan Gonzalez
ethan.gonzalez@utdallas.edu

Thien Nguyen
DangThien.Nguyen@UTDallas.edu

## Abstract

Smart Data Solutions is a strategic partner in healthcare process automation and interoperability that utilizes data and intelligent automation to digitally transform operations and deliver outcomes for our clients, which reduce costs, streamline workflows, and improve overall customer experience. Smart Data Solutions processes and stores hundreds of thousands of healthcare correspondence documents for our customers. Within their repository, clients can search for documents based on individual indexed fields, and they envision a new way for clients to interact with their data and draw intelligence from it. The purpose of the Healthcare Correspondence LLM that we are building is to create a proof of concept for a "chatbot" system that would allow customers to ask questions and receive answers from their correspondence data history, demonstrate how an open LLM architecture can receive correspondence history data, define system and resource requirements for full-blown production system, and create a proposed list of other data integrations that could add value to the system in the future.

## Architecture




- Tesseract OCR
  - Convert image documents containing customer data for over 1000 real-life patients into JSON and TXT files
  - A Python script was used to extract specific JSON values given keys, converting 10,500 files into 10,500 text data files.
- Chroma Database
  - A persistent Chroma database was created, where embeddings of the text files were stored as a file, eliminating the need to regenerate embeddings when new files are added.
- LLAMA 2
  - Using LLama.CPP model optimizer and using GGUP model.
  - Running the model on GPU AWS Workspace and reach up to 15 second processing via Tesla T4 GPU.
- Flask API - Backend
  - Allows the front end to input customer and prompt and feed it to the model as a GET request.
- WebUI Front-end
  - Interacting directly with the chatbot through friendly chatbox WebUI which can be implement anywhere

## Metrics

- 6/6 milestones were achieved, including vectorizing database and integrating LLM with data
- Function exception we met:
  - Chatbot is able communicate with user effectively with probably high accuracy.
- We completed each milestone average in one week
- LLM respondes in 2 ½ minutes for CPU inference, and 15 seconds for GPU inference.

## Impact

- The project allows healthcare experts to ask a chatbot detailed questions about a patient's history, diagnosis based on a collection of documents in order to help patients navigate through documents more easily
- The chatbot is able to answer like a human and provides a convenience to the user as opposed to manually searching or having to learn how to use a 3rd party program
- The website format allows for easy clicking and navigation as well as a visual assistance


## Summary

The project aims to simplify the searching and inference done by healthcare officials by having a customized chatbot that knows the existing data as context, The powerful LLM models available for the public (Llama 2) are able to make powerful inferences based on large contexts windows. The use of ChromaDB allows for a vectorized group of data in order to perform semantic similarity searches, thus avoiding complex tokenization or mapping techniques. Queries are passed to ChromaDB's powerful functions and the output is a parameterizable group of relevant documents.