Kalaiyarasi G

Chennai • kalaiyarasi04ai@gmail.com • 9751718523 • in/kalaiyarasigopal • github.com/kalaiyarasi2

EDUCATION

B.Tech in Artificial Intelligence and Data Science

Annai Mira College of Engineering & Technology - 8.51 • 2021–2025

EXPERIENCE

Machine Learning Intern

Graditwin **February 2025 - May 2025**

- Designed and built an AI document assistant to let users instantly search and query information from their own uploaded files (PDF, DOCX, TXT, CSV, Excel, PPTX) solving the problem of slow and inaccurate manual document review.
- Developed the system in Python, using specialized libraries (PyPDF2, python-docx, pandas, openpyxl, python-pptx) for multi-format text extraction, LangChain for chunking, Sentence Transformers and FAISS for semantic vector search, and Groq API (Llama3-8b-8192) for large language model-based, document-grounded answers.
- Reduced document query response time from minutes to seconds.

AI Developer Intern

Strydo Technologies **June 2023 - July 2023**

- Addressed the challenge of making pre-trained language models work effectively for domain-specific tasks without requiring massive datasets or high-end GPU infrastructure.
- Used Unsloth, a Python library enabling advanced parameter-efficient fine-tuning (PEFT) for large language models Gemma, and Qwen models with comprehensive transformer architecture support.
- Implemented LoRA (Low-Rank Adaptation) adapters targeting key attention parameters (Q, K, V) with 4-bit quantization for significant memory savings and fast training within resource-constrained environments like Google Colab with T4 GPU.
- Successfully fine-tuned multiple large language models with 30x faster training speed and 90% reduced memory usage compared to traditional methods.

PROJECT

Multimodal RAG

July 2025 - August 2025

- Developed a full-stack, multi-modal AI chat application that allows users to upload various file types and ask questions to receive clear answers.
- Text files (PDFs, Word, TXT) extracted using standard libraries like PyPDF2 and python-docx, images analyzed with Google Gemini Vision to extract detailed descriptions, audio files are transcribed using the Deepgram API along with PyDub and soundfile libraries.
- Video files are processed by extracting audio (via moviepy) and key frames (via OpenCV), the audio was transcribed with Deepgram, and frames are analyzed using Google Gemini Vision to generate rich multimodal content.
- The system uses FAISS as a vector database for semantic search, SentenceTransformer’s “intfloat/e5-large-v2” model for embedding generation and chunking to enable efficient retrieval-augmented generation (RAG).
- The frontend built with React and TypeScript provides an interactive chat interface supporting file uploads, creating an easy-to-use app for querying diverse file content.

Real-Time Voice Assistant

June 2025 - July 2025

- Developed a real-time AI assistant named Jarvis to provide natural, clear, and friendly conversational support tailored specifically for personalized user assistance.
- LiveKit SDK for real-time session management (AgentSession, Agent, RoomInputOptions), Google Gemini 2.0 Realtime Model for streaming and integrated cloud-based noise cancellation (BVC) to enhance audio quality.
- Delivered a responsive, human-like AI assistant capable of smooth real-time voice interactions with improved audio clarity and customizable, supportive communication style that enhanced user engagement.

Enhanced Website Summarizer

May 2025 - June 2025

- The project aimed to develop an AI-powered website summarizer that automates extracting and summarizing key content from websites.
- Built the system integrates the LLM powered by llama-3.3-70b-versatile for advanced natural language understanding. It uses the Pydantic AI Agent framework with MCP server(mcp fetch server and firecrawl) integration to manage AI communication asynchronously, ensuring efficient and scalable AI processing.
- Utilized AsyncIO for concurrent website fetching to improve processing speed, Pandas for data manipulation and analysis, and Plotly for creating interactive, insightful data visualizations.
- This addressed the challenge of manual, time-consuming web research by providing users with concise, actionable insights efficiently.

Customer Churn Prediction

April 2025 - May 2025

- Built a Customer Churn Prediction System for telecom data to identify at-risk customers, addressing revenue loss from customer churn by predicting likelihood of churn based on demographic, billing, and service usage data.
- Using Pandas and NumPy for preprocessing, XGBoost Classifier (churn_model.pkl) for high-performance predictions on tabular data, and SMOTE-ENN to effectively balance churn vs non-churn classes; applied robust label encoding with an “Unknown” fallback for unseen categorical values and used Joblib to serialize/load the trained model and encoder (label_encoders.pkl).
- Integrated with a CSV dataset (customer_churn.csv) to auto-fetch existing customer details, enabling real-time churn prediction for business teams.
- The project has potential to reduce churn rates by 10–15% through early customer engagement strategies.

SKILLS

Programming Languages: Python, SQL

Machine Learning Frameworks: TensorFlow, Scikit-learn, PyTorch, LangChain

Data Processing & Data Visualization Tools: Pandas, NumPy, MySQL, PostgreSQL,Power BI, Tableau, Matplotlib, Seaborn, Plotly.

AI Integration & APIs: Model Context Protocol (MCP), Claude AI, Groq API, LiveKit, Google Gemini API (Vision &amp; Realtime), Deepgram, FAISS, SentenceTransformers.

Libraries & Tools: PyPDF2, python-docx, PyDub, soundfile, moviepy, OpenCV, dotenv, SMOTE-ENN

Frameworks & Deployment: Streamlit, Playwright, React, TypeScript, AsyncIO

Gen AI: Fine-tuning & PEFT (LoRA, QLoRA, Unsloth), LLMs (Llama, Gemma, Qwen), RAG with FAISS & SentenceTransformers, Multimodal AI (Gemini Vision/Realtime, Deepgram), AI Agents (Pydantic AI, MCP), Real-time Voice Assistants (LiveKit + Gemini).