Sambhav Khanna

sambhav | sambhavKhanna | sambhav-khanna | sambhav.khanna@uwaterloo.ca | 519-404-4057

Education

University of Waterloo Sep 2022 – Apr 2027
*Bachelor of Computer Science, AI Specialization (Cumulative Average **89.45%**)* *Waterloo, Canada*

- Relevant Coursework: Data Structures and Algorithms, Operating Systems, Numerical Computation, Cryptography, Object-Oriented Programming, Sequential Programs, Computer Architecture, Statistics

Experience

Software Engineer Intern Jun 2024 – Sep 2024
Tallgeese AI | Python, Django, ChromaDB, Langsmith, Ollama *San Jose, United States*

- Implemented accuracy evaluation pipeline to assess **agent** responses, integrating it with GitHub **CI/CD** actions.
- Performed speed and accuracy benchmark tests for Chroma and **Milvus**, and added support for parallel requests.
- Enhanced language support for Chinese by resolving function-calling errors, leading to **77%** increase in accuracy.
- Resolved critical issues in document ingestion and chunking, resulting in a **96%** accuracy improvement and a significant increase in user adoption.

Software Engineer Intern Apr 2024 – Jun 2024
Qatar Airways | Milvus, LlamaIndex, OpenAI API, Flask, Angular.js *Doha, Qatar*

- Developed a **RAG** system for storing **100+** documents, each with **200+** pages containing airplane technical data.
- Indexed the embeddings and stored them in a cloud vector database, to decrease the time to first token by **80%**.
- Implemented data cleaning pipeline, to clean tabular data, thereby increasing the accuracy of responses by **100%**.
- Managed an intern and assisted them in prompt engineering, RAG API development and frontend development.

Full Stack Engineer Intern Sep 2023 – Dec 2023
Toronto Transit Commission | Python, TypeScript, React.js, Electron.js, Docker *Toronto, Canada*

- Developed an encrypted password management application to store and manage **1000+** logins for **40+** databases.
- Reduced the application render time by **80%**, by migrating the codebase from Python **Tkinter** to **React.js**.
- Containerized the application, and used caching to reduce the build time from **5 minutes to 5 seconds**.
- Implemented SQL triggers and transactions to reduce data processing time, by decreasing unnecessary query runs.

Projects

Predictify Pro | *Python, Stripe API, Next.js, Django, GCP*

- Built the website and backend for Predictify Pro, a startup, integrating a RAG chatbot and **Stripe API**.
- Developed an interior design feature that updates room images based on user input by developing **Cycle-GAN**.

ChatrBox | *MongoDB, React.js, Express.js, Node.js, socket.io, nginx, Docker, AWS*

- Implemented a real-time chat server using the **websocket** implementation provided by **socket.io** API.
- Implemented custom hooks and custom contexts using **React Context API** to avoid prop-drilling.
- Developed a client-side routing system using **React-Router-DOM** for smooth transition.
- Containerized the application using **Docker** and deployed the image on an **AWS EC2** instance.

Talk to Doc | *Go, gin, groq, Pinecone, React.js*

- Developed a RAG pipeline with Meta's llama3.1:8b model using **groq** LPU API and backend made with **Gin**.
- Used **Pinecone** vector database to store the document embeddings and **Ollama** embeddings to embed the docs.
- Tested the RAG model with PDF files containing tables and achieved successful response on **94%** of the prompts, asking about data entries of specific cells.

Technical Skills

Languages: Python, TypeScript, JavaScript, Go, C/C++, HTML/CSS
Frameworks: Pandas, Langchain, Express.js, Next.js, Node.js, React.js, Flask, Django, Gin
Developer Tools: Ollama, MongoDB, Firebase, Postgres, MySQL, AWS, GCP, Docker, Git, Linux, Azure