

Thura Win Kyaw

Yuseong-gu Gung-dong, Daejeon, KR

thurawk@gmail.com | github.com/thurawk

Profile

Aspiring AI/ML engineer with hands-on experience in building RAG-powered assistants, developing full-stack AI applications, and designing LLM-based conversational agents. Skilled in transforming research ideas into practical systems.

Research Interests

I am passionate about artificial intelligence, machine learning, and conversational AI, with a focus on Retrieval-Augmented Generation (RAG), chat-bot development, and the integration of vector databases to enhance user interaction and contextual understanding. Committed to developing intelligent systems that deliver accurate, context-aware, and user-centric solutions for real-world applications.

Education

Chungnam National University, BE in Electrical Engineering Sept 2022 – Present

- GPA: 3.35/4.5
- **Coursework:** Modern Control Theory and Design, Sensor and Measurement engineering, Computer Programming, AI and Future Society, Computer Scientific Thinking, Information Technology of Power Systems

Experience

AI, Software, Prompt Engineer, AldenBio – Daejeon, KR Dec 2024 – Present

- Designed and maintained backend systems with a focus on AI model integration, API development, and database management
- Engineered and optimized prompts for Large Language Models (LLMs) to ensure personalized, context-aware responses across various use cases
- Developed and deployed Retrieval-Augmented Generation (RAG) pipelines to enhance the accuracy and relevance of AI-generated outputs
- Contributed to building intelligent, production-grade systems for life sciences and biomedical applications

Projects

ChatPDB – Protein Structure Visualization Assistant (LLM + RAG + PyMOL)

- Developed an intelligent assistant to interpret and visualize protein structures using natural language queries.
- Built with FastAPI, integrated PyMOL for 3D rendering, and used LLM + RAG for semantic query understanding.
- Implemented a retriever pipeline (BM25 + FAISS + Cross-Encoder) with a domain-specific vector store from UniProt annotations.
- Enabled session-based workflows and automated tools (e.g., ligand binding sites, domains, and hydrophobicity).
- Integrated a CLI using Ollama and Qwen/Deepseek LLMs.

Iris Flower Classification with Support Vector Machine (SVM)

- This project involved building a machine learning model to classify iris flowers using a Support Vector Machine (SVM). I used Scikit-learn for model building and evaluation, and visualized model performance with confusion matrices and decision boundaries.

Customer Segmentation with K-means Clustering

- I applied K-means clustering on synthetic customer data to segment customers by age, income, and spending score. The key steps involved data scaling, identifying optimal clusters with the Elbow Method, and evaluating performance with silhouette scores.

Tic-Tac-Toe Game with Python

- Designed a 3x3 grid with dynamic state updates of the game, implemented win, tie, and input validation mechanisms, and ensured seamless player interaction with alternating turns.

Skills

Programming Languages: Python, C, MATLAB.

Tools: Git/GitHub, Google Workspace, Microsoft Office, VS Code, Cursor AI, PyMOL, Jupyter Notebook, Notion.

Libraries & Frameworks:

- Data & ML: Pandas, NumPy, Scikit-learn, TensorFlow, Seaborn, Matplotlib
- LLM & RAG: LangChain, FAISS, Transformers (Hugging Face), SentenceTransformers
- Web & Backend: FastAPI, Flask

AI/ML Techniques:

- Supervised & Unsupervised Learning
- Regression, Classification, Clustering
- Reinforcement Learning
- Retrieval-Augmented Generation (RAG)

Tools & Platforms:

- Development: VS Code, Cursor AI, Jupyter Notebook, PyMOL
- Version Control: Git, GitHub
- Productivity: Notion, Google Workspace, Microsoft Office
- Model Hosting & Inference: Ollama, Hugging Face, Transformers Inference API

Operating System: Windows, Mac, Linux

Languages: Burmese (Native), English (Fluent), Korean (TOPIK Level 5).

References

Available upon request.