Anupam Verma
Engineer 1

- Chennai, Tamil Nadu, India
- 9445864375
- anupam215769@gmail.com
- LeetCode 🔗
- Github 🔗
- Linkedin 🔗
- Portfolio Website 🔗
- Tableau Public 🔗

🧠 SKILLS

Generative AI ●●●●○
RAG, Fine-tuning, LangChain, LangGraph, Transformers

Machine Learning ●●●●○
TensorFlow, PyTorch, Scikit-learn, NLTK, spaCy

Data Analysis & Visualization ●●●●●
Tableau, Power BI

Programming ●●●●○
Python, C++, C

Frontend & Backend ●●●●○
Streamlit, FastAPI, Flask

Databases ●●●○○
MySQL

Cloud ●●●○○
AWS, Docker

Web Automation ●●●○○
Selenium

🎓 EDUCATION

Post Graduate Diploma in Data Science, Symbiosis Centre for Distance Learning
Jul 2023 – Jun 2025

B.Tech in CSE, Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology
Jul 2019 – Jun 2023
Passed with 9.08 CGPA

💼 EXPERIENCE

Comcast, Engineer 1
Jan 2023 – present
Generative AI
- Engineered a **Retrieval-Augmented Generation (RAG)** system with **LangChain** and **LangGraph**, querying millions of code lines and cutting manual reviews by **40%** through automated suggestions, bug detection, test generation, and more efficient documentation of new RDK contributions using **OpenAI GPT-4o** and **o1 models.**
- Developed an interactive **chatbot** with **Streamlit** as the front-end and **FastAPI** as the **RESTful API** back-end, reducing latency by **60%** while creating **custom vector embeddings** through **microservices**.
- Integrated multiple agents with a **ChromaDB** vector store, combining **Similarity and Full-Text Search** to improve retrieval accuracy by **95%** and enable more precise queries of RDK documentation and codebase.
- Implemented intelligent **query routing** to direct inquiries to RDK documentation, codebase, custom codebase, or patch analysis, cutting response time by nearly **25%** and optimizing retrieval based on query intent.
- Integrated a **Neo4j Knowledge Graph** to map and analyze RDK component-level dependencies, accelerating dependency mapping by **40%** and enabling precise identification of inter-component relationships and creation of architecture diagrams.
- Employed **Web Scraping** and **Selenium** to extract details on contributions like Test Procedures and Reasons for Change, leading to correct data extraction by **90%** of the times while downloading patches from Gerrit.
- Subsequently, the results of this analysis were updated in Jira tickets via the **Jira API**, improving cross-team visibility by **70%** and streamlining communication and tracking of code changes with their impact.

Tableau
- Developed **50+** interactive RDK dashboards in **Tableau**, including clone, code, and contribution metrics portals, boosting insights by **75%** and significantly improving data accessibility and efficiency for stakeholders.
- Leveraged advanced Tableau techniques to create visually appealing **charts, graphs, and maps**, representing complex data more effectively and enhancing **stakeholder** understanding by **50%** for greater overall comprehension.
- Implemented innovative data integration with **Tableau Prep Builder** for **ETL** process like cleansing and transformation, cutting data cleaning time by **85%** and improving loading efficiency into **MySQL** while maintaining a streamlined workflow.

📁 PERSONAL PROJECTS

Sentiment Analysis using NLP [TensorFlow | Keras] 🔗
- Developed a sentiment analysis project on Reddit comments using **NLP** techniques, implementing and comparing various neural network models (**Feed-forward, LSTM, GRU, Bidirectional RNN, Conv1D**) to classify comments as "Normal" or "Hate," achieving up to **88.5%** accuracy on the test set—significantly outperforming the baseline accuracy of 77.25%.

Image Classification using Computer Vision [TensorFlow | Keras] 🔗
- Developed an image classification project differentiating animals using **computer vision** techniques. Implemented and compared various models—including **Random Forest** with **HOG** features and **Convolutional Neural Networks** (Tiny VGG, overfitted, reduced overfitting, data-augmented model)—to improve accuracy, achieving up to **72.55%** on the test set.

Movie Recommendation System, [Numpy | Pandas | Scikit-learn] 🔗
- Developed a **content-based** movie recommender system using **cosine similarity** and **K-Nearest Neighbors**. Processed and transformed movie metadata—including genres, keywords, cast, crew, and overview—into feature vectors.

🏅 CERTIFICATES

- Fundamentals of Deep Learning by NVIDIA 🔗
- AWS Academy Machine Learning Foundations 🔗