

ANAND VADGAMA

Rajkot, Gujarat, India | vadgamaanand6@gmail.com | +91 97256 54971 | [GitHub](#) | [LinkedIn](#)

PROFESSIONAL SUMMARY

Final-year B.E. student in Artificial Intelligence & Data Science at GEC Rajkot, actively seeking AI/ML Engineering placement and internship opportunities. Hands-on experience across the full ML lifecycle — data preprocessing, feature engineering, model training, evaluation, and production deployment on AWS. Implemented Transformer and LLaMA 2 architectures from research papers in PyTorch with numerically verified results. Comfortable reading research papers, writing clean production code, and shipping ML systems independently.

EDUCATION

Government Engineering College, Rajkot

B.E., Artificial Intelligence and Data Science · Expected May 2027

CGPA: 7.5 / 10 · No Active Backlogs

TECHNICAL SKILLS

Languages: Python — OOP, data structures, control flow, file handling, modules

ML & Deep Learning: PyTorch, Scikit-learn, NumPy, Pandas, XGBoost, NLTK — ANN, CNN, RNN, LSTM, Transformers, Attention Mechanisms

Algorithms: Linear & Logistic Regression, Decision Trees, Random Forest, SVM, KNN, K-Means, SMOTE

MLOps & Cloud: Docker, DVC, GitHub Actions (CI/CD), AWS EC2, AWS S3, FastAPI

Tools: Git, GitHub, Jupyter Notebook, Google Colab, MongoDB Atlas, Antigravity, Claude Code

PROJECTS

Vehicle Insurance Risk Prediction [GitHub ↗](#)

Python · Scikit-learn · FastAPI · Docker · AWS EC2/S3 · GitHub Actions

- Built an end-to-end supervised ML pipeline (Random Forest) — data ingestion, validation, EDA, feature engineering with SMOTE for class imbalance, training, and evaluation on a real-world tabular dataset
- Deployed a live prediction web app via FastAPI on AWS EC2, with trained models stored and retrieved from AWS S3
- Automated testing and redeployment on every push to main with GitHub Actions CI/CD; containerized the full stack with Docker for consistent dev/prod behaviour

Transformer: English-Hindi NMT from Scratch [GitHub ↗](#)

Python · PyTorch · NLP · Seq2Seq · Attention

- Implemented the complete Transformer architecture in PyTorch from scratch — multi-head self-attention, positional encoding, encoder-decoder stacks — by studying the original research paper
- Built custom English and Hindi tokenizers and an attention-visualisation notebook; tracked BLEU score across epochs; inference supports greedy decoding and beam search

LLaMA 2: Full Architecture from Research Paper [GitHub ↗](#)

Python · PyTorch · LLMs · Advanced Transformer Variants

- Implemented Meta's LLaMA 2 architectural improvements over standard Transformers from the research paper: RoPE, Grouped Query Attention, RMSNorm, KV Cache, and SwiGLU activation
- Loaded Meta's official pretrained weights into the custom build and validated numerical correctness via inference — not just structural similarity

GitHub Activity Analyzer: Modular Python Package [GitHub ↗](#)

Python · REST API · OOP · Matplotlib · Caching

- Designed a clean 6-module Python package (api, analyzer, visualizer, file_handler, utils, main) with single responsibility per module for easy extension and testing
- Pulls any GitHub profile via paginated REST API calls to surface language distribution, commit patterns, and contribution trends; user-specific caching cuts repeat calls, with exports to JSON and high-resolution charts

Additional: 10+ ML/DL notebooks across Healthcare, Finance & NLP (fraud detection, diabetes/cancer classification, spam & fake-news detection) — github.com/AnandVadgama/ML-DL-Projects