# Gowhith Gandem

(214) 776-4433 | gowhith.gandem@stonybrook.edu | https://www.linkedin.com/in/gowhith/ |
https://github.com/GowhithGandem46

## EDUCATION

**Stony Brook University** — Stony Brook, NY
*Master of Science in Data Science* — *August 2024 – May 2026*
- **Coursework:** Distributed Systems, Natural Language Processing

**CVR College Of Engineering** — Hyderabad, India
*Bachelor of Technology in Computer Science and Engineering (AI and ML)* — *August 2020 – July 2024*
- **Coursework:** Operating Systems, Computer Organisation, Computer Architecture, Machine Learning

## TECHNICAL SKILLS

- ***Languages:*** Python, Go, C/C++, Java
- ***Technologies/Tools:*** React.JS, Node.JS, Next.JS, Angular JS, Jenkins, Docker, PyTorch
- ***Databases:*** MySQL, PostgreSQL, MongoDB

## WORK EXPERIENCE

**Software Engineer Intern** — India
*MegaViz Technologies Private Limited* — *December 2023 – May 2024*
- Scaled authentication systems to handle **4,500 logins/sec** by optimizing **PostgreSQL** connection pools and integrating **Bouncy Castle** for token validation.
- Reduced API latency from **200ms to 95ms**, saving **$390,000/year**, by optimizing **RESTful APIs** with **Redis** caching, rate limiting, and **AWS Auto Scaling** for microservices in **Java and Spring Boot**.
- Improved UX to **4.8/5** by building **React** dashboards with dynamic heatmaps and real-time price overlays, reducing UI load times using **React.memo** and lazy loading for **8M+ Users**.

## PROJECTS

**GPT2-124M Implementation**
*Python, PyTorch, Multi-GPU Training, Training Optimization, Performance Engineering*
- Solved on SBU-cluster for **multi-GPU** training **(A100 GPUs)**.
- Coded GPT-2 Base-line(124M) using **PyTorch**, incorporating innovative tokenization strategies to efficiently process nearly 2 million words from the **OpenWebText dataset**.

**Autonomous Vehicle Implementation**
*Python, CARLA, PyTorch, RL*
- Integrated **Deep-RL** into the CARLA simulator by implementing advanced collision avoidance, reducing navigation errors by 25% as measured by path deviation logs.

**Practical Byzantine Fault Tolerance (PBFT) System**
*Go, PBFT Protocol, Goroutines, GRPC*
- Implemented **MIT**-researched **PBFT protocol** with 100% consensus accuracy in adversarial environments.
- Achieved state replication across **7** nodes with **sub-80 ms** latency, supporting **500** concurrent transactions.

**KeyValue Store with RAFT Algorithm**
*Go, TCP/IP, Net/RPC, Multi-threading*
- Developed a sharded key-value store uses **RAFT consensus algorithm**, with Pre-voting, Log Compaction, Log Persistence and Distributed Transactions.

**Summarization Tool**
*Python, Transformers, PyTorch, PyWebio*
- Modified Google's T5 Transformer to develop a text summarization tool, improved summary accuracy by 25% from 65% to 90%.
- Programmed a user-friendly interface with PyWebio, facilitating seamless interaction with summarization tool.

## ACHIEVEMENTS

- 1st in Grand Finals Smart India Software Edition Hackathon with Prize Money $**1200**.
- Won Square's Build What's Possible Hackathon for enhancing shopping efficiency with Square's Invoices API.
- Secured first place in 'Hack the Inevitable' hackathon by architecting a novel blockchain interoperability solution, increasing transaction speeds by 40% and reducing cross-network fees by 15%.