

Shashwat Narayan

+91 8102116436 | shashwatnarayan15@gmail.com | linkedin.com/shashwat | github.com/shashwat

PROFESSIONAL SUMMARY

M.Tech candidate in Computational Data Science with a strong mathematical foundation in **Linear Algebra** and **Stochastic Processes**. Focused on bridging the gap between Statistical Learning and Deep Neural Networks to engineer **latency-critical systems**. Proficient in building **real-time decision architectures** that maximize **algorithmic throughput** on constrained hardware using Python and C++.

EDUCATION

National Institute of Technology (NIT)

M.Tech in Computational Data Science

Surathkal, Karnataka

Aug. 2025 – Present

Indian Institute of Technology (IIT)

B.Tech in Textile Technology

New Delhi, India

Nov. 2020 – May 2024

PROJECTS

NLP Classifier Benchmarking Platform | *Python, HuggingFace, ONNX, Docker, FastAPI*

- Built a hardware-aware benchmarking engine to compare Statistical (Naive Bayes) vs. Neural (DistilBERT) architectures, implementing dynamic GPU/CPU switching via ONNX Runtime.
- Quantified the latency-accuracy frontier: Achieved 98.5% F1-score with DistilBERT (GPU) while identifying Naive Bayes as 1,320x faster (0.009ms latency), validating statistical methods for high-frequency packet filtering.
- Optimized Transformer inference latency by 60% on CPU and achieved 4.5ms latency on RTX 3050 GPU using ONNX quantization and execution providers.
- Deployed a production-ready FastAPI microservice to Docker Hub, featuring custom middleware for real-time latency metrics and SMOTE synthetic data generation.

High-Performance Image Quantization SaaS | *Python, Streamlit, Numba, Docker*

- Architected a full-stack image segmentation app using a custom K-Means engine, utilizing Numba JIT compilation to accelerate distance calculations by ~287x compared to standard iterative Python.
- Engineered a “Train Small, Apply Big” optimization strategy that performs centroid training on adaptively downsampled data, enabling the processing of 4K images in under 3 seconds.
- Developed a rigorous computational analysis module to benchmark algorithmic efficiency (Loops vs. Vectorization vs. JIT) and automate hyperparameter tuning using the Elbow Method.
- Containerized the application using Docker, ensuring reproducible runtime environments and dependency isolation for cloud-agnostic deployment.

Heuristic Search Algorithm Analysis Platform | *Python, Pygame, Data Structures*

- Engineered an interactive simulation engine to visualize and benchmark pathfinding algorithms (A*, Dijkstra, BFS) on weighted and unweighted graphs using Object-Oriented Programming (OOP).
- Implemented custom heuristics (Manhattan distance) and priority queue structures to optimize search efficiency, reducing search space by over 60% compared to blind search methods.
- Integrated weighted node logic to simulate complex traversal costs, visually demonstrating the divergence between cost-based (Dijkstra) and step-based (BFS) optimization.
- Developed a real-time benchmarking system to calculate and display computational cost (nodes visited) and path optimality dynamically during execution.

TECHNICAL SKILLS

Languages: Python, C, SQL, Bash Scripting

High-Performance Computing: Numba (JIT), ONNX Runtime, Multi-threading, Vectorization

Frameworks & Libraries: FastAPI, Streamlit, HuggingFace Transformers, NumPy, Pandas, Scikit-learn, Pygame

DevOps & Tools: Docker, Git, VS Code, Linux/Unix Shell, Jupyter Notebook

Core Concepts: Data Structures & Algorithms, Graph Theory, Heuristic Optimization, Stochastic Processes, NLP, System Design