Ayush Naique

+1-2012342533 | asn9772@nyu.edu | ayushnaique28 | ayushnaique

Education

New York University - Courant Institute of Mathematical Sciences — New York, NY
Masters in Computer Science, CGPA: 3.89/4 — *May 2025*

National Institute of Technology, Goa — Goa, India
B.Tech in Computer Science and Engineering, CGPA: 8.99/10 — *May 2023*

Technical Skills

Languages: Python, C/C++, OCaml, Scheme, Java, SQL, MongoDB, JavaScript, HTML/CSS, Scala, Ada
Frameworks: React, Node.js
Tools and Libraries: Git, VS Code, Visual Studio, Jupyter, pandas, NumPy, Matplotlib, SciPy, Tensorflow, Scikit learn
Relevant Courses: Algorithms, Operating Systems, Compiler Construction, Computer Vision

Experience

National Institute of Technology, Goa — Sept. 2021 – May. 2023
Undergraduate Researcher — *Goa, India*

- Developed a specialized speech detection system for classifying Indian dialects.
- Implemented robust classification models leveraging phonetic and spectral features from Konkani and Kannada datasets.
- Employed advanced embeddings like Wave2Vec and X-Vectors for feature extraction and achieved notable accuracies of 86% on LSTM and 88% on ANN models, surpassing benchmark SVM accuracy by 4%.

National Center for Polar and Ocean Research — Jun. 2022 – Sep. 2022
Research Intern - Machine Learning — *Goa, India*

- Deployed Deep Learning models for polar weather forecasting at the Bharati Antarctic Research Station, focusing on LSTM, RNN, and Sequence2Sequence architectures.
- Investigated seasonality trends in time-series data using advanced techniques like PCA and discriminant analysis for feature extraction and seasonal effects mitigation.
- Expertise in parameter optimization led to significant improvements in model accuracy. Achieved outstanding performance with LSTM and Seq2Seq-based LSTM models, including an RMSE of 0.009 and MAE of 0.006.

Smartknower Ltd — Jan. 2022 – Mar. 2022
Data Science Intern — *Bangalore, India*

- Implemented and optimized clustering algorithms for robust classification model development.
- Utilized Census Database for dataset curation, facilitating effective categorization of the Indian population based on credit scores and related factors.
- Improved the prediction models using Bayesian Classifier and Random Decision Forest algorithms in Python, leveraging libraries such as Scikit-Learn and SciPy for efficient implementation and analysis.

Projects

Compile-time Optimizations for Iteration and Recursion | OCaml

- Specializing in optimizations throughout the compilation pipeline, particularly targeting iteration and recursion during code generation.
- Implementing strategies to recognize and optimize tail recursive functions by applying Tail Call Optimization.
- Planning to incorporate techniques such as loop invariant removal and loop unrolling to further enhance performance and efficiency.

Multimodal Image Colourization using Diffusion | Python, PyTorch

- Focused on leveraging diffusion models to optimize image colourization processes and bolster their self-learning capabilities.
- Exploring innovative architectural modifications and integrating diverse multimodal features, such as incorporating video clips as input data.
- Conducting comprehensive data pre-processing and augmentation to enrich the training dataset. Responsible for overseeing the training and inference procedures of diffusion-based image colourization models.

Operating System Simulation | C++, STL

- Engineered a two-pass linker enabling the consolidation of code modules into a unified executable, resolving external symbol references, and practising module-relative addressing.
- Evaluated diverse process scheduling algorithms (FCFS, LCFS, SRTF, Round Robin, Priority, Preemptive Priority) using Discrete Event Simulation techniques.
- Designed a robust Memory Management Unit (MMU) facilitating virtual-to-physical address mapping. Integrated advanced page table translation mechanisms and supported various page replacement algorithms (FIFO, Clock, NRU, Aging, Working Set).

Relocating SIC Assembler | C++/C, STL

- Created a relocatable program assembler for a 20-bit Simple Instructional Computer (SIC) in C++, capable of generating object code from machine instructions.
- Operated using Object-Oriented Programming principles to structure the assembler tables and enhance modularity.