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 — *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++, Java, SQL, MongoDB, JavaScript, HTML/CSS, OCaml, Scala, Ada, Scheme
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 Design, Computer Vision

Experience

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

- Developed Deep Learning models for prediction of polar weather for the Bharati Antarctic Research Station.
- Implemented Deep learning models using Time Series data-set, mainly Long Short-Term Memory (LSTM), RNN and Sequence2Sequence models.
- Conducted relevant research on the effects of seasonality trends in time-series data for the ARIMA and LSTM prediction models. Used PCA, discriminant analysis to carry out feature extraction and got rid of seasonality from the data.
- Tested, analyzed and identified parameters that increased the overall accuracy of the learning model. Collaborated with fellow interns to determine optimal prediction models.
- Achieved best performance for LSTM and Seq2Seq-based LSTM models with an RMSE value of **0.009** and MAE of **0.006**.

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

- Implemented and optimized clustering algorithms for creating classification models.
- Worked on Census Database for building the data-set for classification. Classified the Indian populace using their Credit Scores and other Credit related details.
- Prediction model developed using Bayes Classifier and Random Decision Forest based algorithms using Python with libraries such as Scikit-Learn and SciPy.

Projects

Indian Dialect Classifier — Sep. 2021 – May 2023

- Speech detection system based on the dialect of Indian language.
- Dialect classifier modeled on Artificial Neural Networks, Support Vector Machine, Ensemble Models and Deep Neural Network.
- Extracted phonetic and spectral features from the Konkani and Kannada language dataset. Made use of embeddings such as Wave2Vec, X-Vectors and Bottleneck Features to extract suitable features for training of the classifier models.
- Attained **86%** accuracy on LSTM model and **88%** accuracy on ANN model. A significant increase in accuracy was found from the benchmark classification model using $3/4^{th}$ of the audio samples (SVM on MFCC features: **82%**)

Operating System Simulation — Sep. 2023 – Dec 2023

- Simulated Operating System processes like Linkers, Schedulers, Memory Management Unit and Disc Scheduling.
- Implemented a two-pass linker which takes individually compiled code/object modules and creates a single executable by resolving external symbol references and module relative addressing by assigning global addresses after placing the modules' object code at global addresses.
- Utilized process scheduling algorithms like FCFS, LCFS, SRTF, RoundRobin, Priority, and Preemptive Priority. Utilized Discrete Event Simulation to assess scheduling performance.
- Designed an MMU to map virtual address spaces of multiple processes onto physical frames using page table translation. Page replacement algorithms like FIFO, Clock, NRU, Aging, Working Set, etc. were supported.

Relocating SIC Assembler — Jan. 2021 – May 2021

- Implemented an Simple Instructional Computer(SIC) assembler for 20 bit relocatable programs in C++. When given a machine instruction as an input, the SIC Assembler outputs the object code.
- Used Object Oriented Programming protocols to design the assembler tables.

P2P Communication Applet — Jan. 2021 – May 2021

- Created an end-to-end communication link using Java.
- Made use of Java multithreading to allow multiple users to communicate at once. Used Java libraries such as Swing to design the applet.
- Used Computer Networking designs and Java sockets to establish communication link between clients and server. Achieved performance of **30** concurrent users without significant performance drop.