

Akshay Nayak

akshay.nayak@gwu.edu

571-622-9592

github.com/kayanyahska

linkedin.com/in/kayanyahska

[Google Scholar](#)

Education

George Washington University

Washington DC, USA

Master of Science in Computer Science (Thesis) (GPA: 3.83/4.00)

August 2024 – May 2026

- Relevant Courses: Advanced Software Paradigm, Design & Analysis of Algorithms, Machine Learning, Computer Architecture, Neural Networks & Deep Learning, Trustworthy AI.

National Institute of Technology, Warangal

Warangal, India

Bachelor of Technology in Computer Science and Engineering (GPA: 7.23/10.00)

July 2017 – May 2021

Technical Skills

- **Programming Languages:** Python, C++, Java, C, Golang, JavaScript, TypeScript.
- **Databases:** MySQL, MongoDB, PostgreSQL, Vector Databases, Redis.
- **Frameworks & Libraries:** Django, Flask, ReactJS, Next.js, TensorFlow, PyTorch, Keras, Scikit – Learn, NodeJS, OpenCV, Numpy, Pandas, Drizzle ORM, Tailwind CSS.
- **AI/ML & Cloud:** RAG (Retrieval Augmented Generation), Vector Search, Embeddings, Vercel AI SDK, Semantic Search.
- **GPU & Parallel Computing:** CUDA, cuDNN.
- **Tools:** Linux, Docker, SQL Server Management Studio (SSMS), Git, JIRA, Confluence, Stash, Inngest, Vercel.

Work Experience

Mastercard

Pune, India

Software Engineer II

July 2021 – August 2024

- Architected a multi layered analytics versioning feature, that boosted operational efficiency of our clients by 65%. This system empowered users to trace and compare modeling strategies across versions, trace parameter impacts on resultgrids, trendcharts, segmentation views, and predictive models directly contributing to doubling annual client renewal rates.
- Involved in designing and building the back-end system, A / B testing feature in Python, ReactJs, and Django to accurately measure the impact of experiments by comparing the segments of the matched test and control group in financial metrics, geography, and customer spend, allowing scalable decision analysis.
- Coded and maintained production features as part of the core engineering team for Mastercard's On Premise Test & Learn® platform (B2B), rolled out to 10+ APAC clients. Contributed across backend (Django, PostgreSQL) and frontend (React) to support enterprise-scale experimentation.
- Resolved a critical performance bottleneck by slashing a key data processing task's execution time from over 2.5 hours to under 3 minutes. This was achieved by re-engineering the underlying Python code and data handling, directly improving platform usability and client satisfaction.
- Solved persistent, critical memory allocation failures in the modeling pipeline that impacted enterprise clients. The solution involved implementing a data chunking strategy, ensuring platform stability and scalability for the SGDSRegressor algo on large datasets containing 10+ millions rows.
- Contributed directly to a 120% product revenue increase by designing, building, and delivering 7+ full stack features, including an interactive driver summary dashboard using React and Redux that provided clients with actionable, at a glance insights.
- Implemented 'Attribute Bucketing' feature from concept to deployment, delivering a crucial data preprocessing tool for ML models. Developed the intuitive frontend UI in React and integrated it with robust backend Python REST APIs.
- Boosted application reliability by building a comprehensive React and Jest unit testing suite, increasing test coverage across all critical components.
- Drove a 45% increase in customer engagement by applying advanced statistical models (linear regression, clustering, segmentation) to user data. The insights were used to design and implement new data driven features that enhanced the product's predictive capabilities.
- Served as one of the primary on call technical resource on rotational basis for the Test & Learn platform, rapidly resolving urgent production issues to ensure high availability and platform reliability.
- Mentored 5+ junior engineers and interns on the platform's codebase, providing guidance on advanced debugging, problem-solving strategies, and established engineering best practices.

Research Experience

George Washington University (Advisor: [Prof. Alexa Alice Joubin](#))

Washington DC, USA

AI Developer Intern

May 2025 - August 2025

- Assembled and deployed a complete RAG as a Service platform using Next.js 14 App Router, TypeScript, and PostgreSQL with Drizzle ORM, demonstrating ability to design scalable full stack applications.
- Optimized semantic search capabilities using vector embeddings and implemented RAG system with Vercel AI SDK, enabling contextually

accurate AI responses based on document content.

- Established database schema with chat file relationships and launched multi tenant system allowing isolated AI contexts, while building real time chat interface with message persistence, file citations, and background job processing using Inngest for scalable document processing.

Alan Turing Institute (Advisor:[Prof. Ganna Pogrebna](#))

London, UK

Machine Learning Researcher

January 2021 – July 2021

- Applied LDA to analyze differences in customer perceptions between finance and fintech companies, identifying key factors influencing satisfaction from online reviews. Used XLNet to integrate company attributes with context-aware sentiment analysis, uncovering deeper insights from unstructured feedback.
- Created a benchmarking framework to evaluate customer satisfaction across finance and fintech sectors, implementing guidelines that optimized workflows and improved service standards by 55%.

National Institute of Technology, Warangal (Advisor:[Prof. M. Srinivas](#))

Warangal, India

Undergraduate Research Assistant

July 2019 – August 2020

- Formulated a neural network to detect COVID-19 infections from chest X-ray images, achieving 95% accuracy with INASNET.
- Engineered methods to identify altered or forged files and uncover steganographic content using ResNet50 and VGG16.

Computational Intelligence Lab, Indian Institute of Science (Advisor:[Amarjot Singh](#), CEO, SkyLark Labs, & [Prof. Omkar](#))

India

Research Intern

May 2018 – July 2018

- Pioneered a novel system for identifying disguised individuals by leveraging facial landmarks and motion signatures. Developed a comprehensive dataset comprising 3,072 videos, featuring 16 disguises, 2 motion patterns, and 6 positional variations for each subject.
- Applied MTCNN for precise facial landmark detection and Part Affinity Field for binary stick frame video generation, enhancing motion-based analysis.
- Architected a pose estimation and identification model based on the 3D-ResNet framework, achieving precise classification across 16 unique classes. The dataset features 192 videos per class, encompassing diverse activity patterns from 8 men and 8 women aged 17–21.

Publications

- Akshay Nayak, Pruthvi Taranath, "Behavioral Data Science in financial services", forthcoming in Pogrebna, G. & Hills, T. (2025) eds. Handbook of Behavioural Data Science, Cambridge University Press.
- Murukessan Perumal, Akshay Nayak, et.al. "INASNET: Automatic identification of coronavirus disease (COVID- 19) based on chest X-ray using deep neural network", ISA Transactions, 2022.
- Dr. M Srinivas, S. Saurav, Akshay Nayak, and Murkeshan A P. "Multi Features Based Facial Expression Recognition", published in Machine Learning Algorithms and Applications book chapter, Wiley Press, 2020.

Projects

Supervised Fine-Tuning of LLM for Code Generation

- Fine tuned Deepseek Coder 7B on 5,000 programming problems from the Hugging Face APPS dataset to improve task-specific reasoning and improving Python code generation.
- Designed a complete SFT pipeline using PyTorch & Hugging Face Transformers, including dataset preprocessing, model loading, tokenizer setup, and training loop customization.
- Integrated manual evaluation, early stopping, and LR scheduling with visualization of training dynamics (loss vs epoch, LR curve).
- Optimized training performance with gradient checkpointing, and memory-efficient batching.

Go Audio Visualizer Tool

- Conceptualized Go-based solution to address performance limitations in audio signal analysis tools like Librosa.
- Built a web application using the Gin framework, Bootstrap 4, and jQuery, enabling users to process mono and stereo .wav files through an intuitive interface.
- Executed STFT with the GoSSP library and optimized computations using Go Routines, enhancing runtime efficiency.
- Visualized time-frequency spectrograms using GNU Plot, showcasing expertise in signal processing and data representation.

Veritas Core

- Constructed a Java based simulator for a custom 16-bit CISC instruction set architecture (ISA), incorporating an assembler, single port memory, and a 16 line FIFO cache to emulate ISA operations and streamline machine code execution.
- Enhanced memory management using a direct mapped single port memory system and a 16-line FIFO cache, ensuring efficient data retrieval and accurate ISA simulation.
- Crafted a real time user interface with live register displays, instruction decoding, and builtin error detection, enabling efficient debugging.