Nirvek Pandey

San Diego , CA · nipandey@ucsd.edu · (562) 367 - 5538 · https://nirvekpandey.com · US Citizen

Education

University of California San Diego **La Jolla, CA**
B.S./M.S. in Computer Science, *Regents' Scholar* September 2022 – June 2027

- **Fundamentals:** Object Oriented Design, Computer Organization, Advanced Data Structures, Computer Security, Theory of Computation, Design and Analysis of Algorithms, Software Engineering, Database Systems
- **AI and ML:** AI: Search and Reasoning, AI: Probabilistic Models, ML: Learning Algorithms, Deep Learning, Natural Language Processing, Recommender Systems and Web Mining
- **Systems and GPU:** Operating Systems Principles, Networked Services, Parallel Computing, Graduate OS (WIP)

Experience

Data Science Intern **Costa Mesa, CA**
Newton(Neurolens) June – September 2024

- Collaborated with data architects to implement an ETL system using SQL and Azure Pipelines, separating analytics workloads that improved pipeline reliability by 27% and accelerated R&D analysis.
- Built an ML pipeline with OpenCV and PyTorch to detect suppressed eye-tracking signals with 84% accuracy in identifying suppressed measurement and enabling more reliable data interpretation for optometry research.
- Created data visualizations utilizing Matplotlib and Seaborn to illustrate the impact of architectural updates.
- Influenced stakeholder decisions to invest in regression-based modeling for next-generation device development.

Project Manager Intern **La Jolla, CA**
UC San Diego, CSE 110 April – June 2024

- Led a cross-functional Agile Scrum team of 8, facilitating weekly stand-ups and retrospectives that enhanced communication and reduced update turnaround times 46%, managing task progress using GitHub Projects.
- Designed CI/CD workflows with GitHub Actions, reducing deployment errors by 24% and accelerating releases through automated JUnit unit tests and Jest/Puppeteer-based integration and end-to-end testing.
- Directed sprint planning and backlog refinement to synchronize front-end and back-end efforts, increasing development velocity by 20% and raising milestone delivery rates from 65% to 85% over two quarters.

Projects

Personal Portfolio *Next.js, Go, Firebase, PostgreSQL, Google Cloud Run* January 2025 – Present

- Upgraded full-stack portfolio from static React to Next.js + Go microservice architecture, implementing SSR, batched loading, and CDN caching to achieve 75% fewer API calls and 40% faster load times.
- Integrated Google Drive API for dynamic, location-tagged content management, enhancing personalization while maintaining high availability through automated CI/CD with Docker, GitHub Actions and Vercel.

TritonTube *Go, SQLite, gRPC, etcd, AWS EC2, FFmpeg, MPEG-DASH, HTTP* May – July 2025

- Engineered fault-tolerant video platform with RESTful APIs for upload/playback, implementing MPEG-DASH adaptive streaming to optimize video quality and reduce buffering by over 50% across varying network conditions.
- Built scalable metadata and content services on AWS EC2 with etcd-backed Raft consensus protocol, ensuring data consistency and high availability while supporting 100+ concurrent users for video processing workloads.

Research - RedShift *CUDA, HuggingFace, Ollama, Prompt Engineering* January – March 2025

- Enhanced an automated red-teaming framework to evaluate large language model jailbreak vulnerabilities, expanding detection efforts to over 2,000 attacks across 50 adversarial prompts.
- Built an ML evaluation pipeline where LLMs served as attackers, defenders, and judges, drawing on the "Distract LLMs for Automatic Jailbreak Attack" framework to measure jailbreak success evaluation.
- Standardized and preprocessed adversarial prompt datasets applying custom Python/PyTorch scripts , enabling consistent benchmarking of jailbreak vulnerabilities across major models, with metrics monitored via WandB.

Blackjack Optimizer *Python, Pygame, NumPy, Pandas, SciPy* May 2024

- Developed a RL agent for Blackjack using Q-learning and gradient descent within a Markov Decision Process.
- Iteratively grew the win rate from 23% to 45% over thousands of simulated games.
- Adding online multiplayer support with Flask and Socket.io for real-time gameplay with up to six players.

Skills

Programming Languages: Python, Go, Java, JavaScript/TypeScript, C, C++, SQL, HTML, CSS
Libraries & Frameworks: PyTorch, NumPy, Pandas, Matplotlib,TensorFlow, Scikit-Learn, OpenCV, React, Next.js, Angular, Express.js, Node.js, Flask, Tailwind CSS, JUnit, Jest, Puppeteer, FFmpeg

Tools & Technologies: Git, Docker, gRPC, etcd, PostgreSQL, SQLite, MongoDB, Supabase, Firebase, Google Drive API, Google Cloud Run, Vercel, Azure Pipelines, AWS EC2, AWS S3, WandB, CUDA, OpenCL, MPEG-DASH, GitHub Actions, Linux, Figma, SharePoint, Excel, PowerPoint