Jenish Vikrambhai Patel

New Brunswick, NJ | 201-300-7868 | jenishvpatel4995@gmail.com | linkedin.com/in/jenish–patel | github.com/JenishPatel20 | Portfolio

Education

Rutgers University — New Brunswick, NJ
M.S. in Computer Science, GPA: 3.75/4.0 — *Expected Dec 2026*
Coursework: Machine Learning, Algorithms, Database Systems for Data Science, Massive Data Mining, Intro to AI, Brain-Inspired Computing

Rutgers University — New Brunswick, NJ
B.S. in Computer Science, Minor in Business Administration, GPA: 3.75/4.0 — *May 2024*
Coursework: Intro to Data Science, Software Methodology, Discrete Structures I & II, Intro to Finance

Technical Skills

Languages: Python, JavaScript, TypeScript, Java, C++, SQL, HTML/CSS
AI/ML & Data: PyTorch, scikit-learn, XGBoost, SHAP, TensorFlow/Keras, pandas, NumPy, matplotlib, Jupyter
Web/Backend: FastAPI, Node.js, Express.js, REST APIs, React, Next.js, Tailwind CSS
Databases/Tools: PostgreSQL, MySQL, IndexedDB, Firebase, S3, Git, GitHub, Docker, Gemini API, Anthropic API, WebGPU, WebLLM

Experience

AI/ML Engineer Intern | *Greenlit* | *Co-op* — *Remote | Jan 2026 – Present*

- Developed a **20+ endpoint FastAPI market intelligence service** backed by PostgreSQL, supporting movie search, sentiment timelines, genre health scoring, comparable-film lookup, and cast/people analytics.
- Built a configurable scraping and enrichment pipeline using **Python, BeautifulSoup, Gemini API, TMDB API, and S3** to extract, structure, store, and enrich entertainment-market data.
- Implemented Gemini-powered analysis workflows that extract themes, tropes, tone, originality signals, audience reception predictions, and concise investment briefs from film loglines and comparable data.
- Designed analytics features including weighted **0–100 genre health scoring**, YouTube sentiment aggregation, and a Vanilla JS dashboard with **8+ Chart.js visualizations**.

Building Operations Manager | *Rutgers University* | *Part-Time* — *New Brunswick, NJ | Sep 2021 – May 2024*

- Coordinated daily building and event operations by assigning tasks, prioritizing urgent requests, guiding student staff, and improving communication through Microsoft Teams workflows.

Product Management Fellow | *Blueprint* | *Fellowship* — *New Brunswick, NJ | Jan 2023 – Apr 2023*

- Collaborated with a student product team to define user needs, write user stories, conduct competitor research, and support roadmap planning and final presentation materials.

Projects

Auction Master | Full-Stack Auction Marketplace — GitHub

- Contributed major full-stack development to a team-built auction marketplace using **Node.js, Express.js, MySQL, Firebase, and Vanilla JavaScript**, supporting Admin, Customer Representative, and Buyer workflows across listing creation, bidding, auction resolution, and order history.
- Implemented MySQL-backed bidding logic, reserve-price handling, image uploads with **Multer**, and preference-based email alerts with **Nodemailer** that matched new listings against saved buyer categories and price ranges.

Edge AI LLM Benchmark | Browser-Native LLM Performance Tool — GitHub

- Contributed across a browser-native LLM benchmark using **WebGPU, WebLLM, IndexedDB, and Vanilla JavaScript**, enabling open-source models to run locally in-browser without backend inference servers or API cost.
- Built benchmarking workflows for **TTFT, tokens/sec, memory usage, thermal throttling, MMLU scoring, CSV export, and cloud API comparison**, observing Llama 3.2 1B at **73–172ms TTFT and 48–52 TPS**.

Customer Churn Prediction | Explainable ML Pipeline — GitHub

- Built an explainable ML pipeline in **Python, scikit-learn, XGBoost, SHAP, and pandas** across Telco, SaaS, and BankChurners datasets, training **9 classifiers** and achieving **AUC $\geq$ 0.85** on Telco and BankChurners.
- Generated SHAP plots, ROC curves, and confusion matrices to identify churn drivers such as contract type, tenure, credit utilization, and transaction frequency.

Industrial Fault Detection via SNNs | Spiking Neural Networks Research — GitHub

- Owned a surrogate-gradient **Spiking Neural Network** implementation in PyTorch, using rate coding and differentiable spike approximations to train on industrial vibration and ECG time-series anomaly data.
- Processed CWRU `.mat` vibration files with SciPy into **32×32 windows** across 10 fault classes and supported ECG preprocessing through normalization, padding/truncation, and reshaping.

Belief-State Robot Localization | AI Search and CNN-Guided Planning — GitHub

- Collaborated on a robot localization system using **Python, PyTorch, BFS, and CNN regressors**, encoding belief states as **6-channel spatial tensors** to predict localization cost under uncertainty.
- Compared greedy, CNN-guided, and dual-CNN policies across belief sizes 1–20, with learned policies outperforming greedy baselines on larger belief sets.