Avi Aggarwal

972-984-8921 | aggarwal.avi@gmail.com | **aviaggarwal.org** | **linkedin.com/avi-aggarwal** | **github.com/ObviAvi**

Education

Purdue University — Lafayette, IN
B.S. in Computer Science, GPA: 3.93 — *Expected May 2028*

Experience

IBM — San Jose, CA
Software Engineering Intern, WatsonX Data Team (Summer 2026) — *May 2026 - August 2026*

- Extended the Open-Source OpenSearch Kubernetes Operator with a fault-isolated status reconciler that polls live diagnostic APIs and publishes cluster health as a native ConfigMap object, enabling SREs to replace dozens of manual API calls with a single kubectl get command when diagnosing cluster issues.
- Re-implemented the Kubernetes local-path-cleaner provisioning tool from Python to Go as a statically compiled binary, reducing container image size 94% (55MB → 3.5MB) and startup latency 90% (300ms → 30ms) while eliminating runtime dependencies

Google Developers Group Project Manager — Lafayette, IN
boiler-check.vercel.app — *Feb 2026 – May 2026*

- Led a team of 8 developers to create a RAG-based AI assistant that helps users check their questions against 4000+ official Purdue University policies and documents to get accurate, context-specific responses with the relevant sources
- Built a web scraping pipeline to collect text and image data from Purdue University policy websites and documents, implemented a RAG architecture for multi-modal information retrieval and generation using LangChain and Pinecone, benchmarked different kinds of retrieval systems and embedding models (i.e. vanilla, reranker, etc) to determine the best speed vs accuracy trade-offs, added live response streaming as well as hosted and open-source embedding model endpoint for low latency and better user experience

Promega — Madison, WI
Applied ML Intern (Summer 2025) — *May 2025 – August 2025*

- Designed a machine learning and statistical anomaly detection system integrating environmental sensor data to identify conditions linked to low-quality experimental results and improve instrument reliability. Processed uni-variate and multivariate data in Python with cyclical time encoding, performed correlation analysis, and trained ML based: Holt-Winters, K-means Clustering, Isolation Forest models.
- Integrated an SHT45 sensor into Discover LLC boards with firmware (C/C++) and hardware modifications as well as updated host software to create a pipeline that can log and process environmental readings in real time.

Data Mine Research (Corporate Partnership with Knudsen Institute) — Lafayette, IN
Purdue University — *August 2024 – May 2025*

- Fine-Tuned Named Entity Recognition (NER) models on manufacturing language to improve accuracy of Natural Language Processing (NLP) systems that interpret manufacturing-related communication, especially during surge environments. Additionally, integrated an active-learning pipeline that allows further refining of the model, particularly in its weakest areas.
- BeautifulSoup and Selenium were used to extract unstructured data in HTML and XML files from manufacturing websites, while PyTorch and HuggingFace were utilized for model development and architecture.

Research

Undergraduate Researcher – AI Agent & Benchmark Evaluation (Advised by Dr. Tan) — Jan 2026 – Present

- Analyzed failures of 7 state-of-the-art coding agent scaffolding (Live-SWE-Agent, Mini-SWE-Agent, etc.) on 367 SWE-Bench Verified tasks, identifying 19 universally failed tasks requiring $\leq$ 10-line patches. Showed that factors such as overspecified test-suites and creating symptom-level patches is a primary failure driver over simply patch length or model weaknesses.
- Developed the *Spec-Test Coverage Gap (STCG)* metric with a 4-point rubric and dual LLM-as-judge evaluation pipeline (Gemini 2.5 Pro, Claude Sonnet 4.6), calibrated against expert annotations to quantify test-suite overspecification as a metric to evaluate benchmarks.

Personal Projects

Clariti — clariti demo

- A smart memory-sharing and recall app for people living with dementia. It turns photos, voice notes, and written memories into a living, searchable library. Includes voice Q&A with RAG to retrieve relevant memories, facial recognition for person matching, multimodal image descriptions, and database architecture for family groups.

Machine(Learn); — machine(learn) demo

- Automated ML project which uses agent swarms to train models and find the best solution for a given ML task. Includes an approach planning agent, implementation and hyperparameter tuning agent swarm, and reporting on Modal GPU infrastructure with a Supabase-backed realtime dashboard for dataset upload and live run updates.

Technical Skills

Publications: Promega-Connections

Languages: Python, Java, C, C++, Go, HTML

Frameworks: PyTorch, Hugging Face, Scikit-learn, NumPy, Pandas, React, Next.js, Node.js, Express.js, FastAPI, Expo, LangChain

Developer Tools: MongoDB, Supabase, Firebase, Pinecone, OpenSearch, Kubernetes, Modal, Git, BeautifulSoup, Selenium

Concepts: Machine Learning, RAG Systems, Agent Orchestration, Distributed Systems, Data Structures & Algorithms, Software Design Patterns, Kubernetes, Databases, System Design, Competitive Programming, IoT & Embedded Systems