Kushal Adhyaru

+1-408-748-6342 | kushalatulbhai.adhyaru@sjsu.edu | linkedin.com/in/kushal-adhyaru | github.com/kushal511
San Jose, California - 95113, USA

Education

San Jose State University — California, USA
Master of Science in Data Science with a specialization in Data Engineering; GPA: 3.5/4.0 — *August 2024 - May 2026*

Indus Institute of Technology & Engineering — Gujarat, India
Bachelor of Technology in Computer Science; GPA: 4.0/4.0 — *June 2019 - May 2023*

Skills

Programming: Python, SQL, Java, C++, HTML, CSS, JavaScript

Frameworks/Tools: ReactJS, Node.js, Express.js, Flask, Docker, Kubernetes, Kafka, Redux

Machine Learning, Deep Learning & NLP/LLMs: scikit-learn (modeling), MLflow (experiment tracking), TensorFlow/PyTorch (deep learning), NLTK, Hugging Face Transformers, LangChain (RAG), Flask (web development).

Big Data, Databases & Cloud: Hadoop/Spark (distributed computing), ETL (Prefect, Dagster), MySQL (Relational Database), MongoDB (NoSQL Database), AWS (cloud deployment)

Experience

Software Development Engineer Intern — May 2025 – August 2025
Amazon — *Seattle, USA*

- Implemented an end-to-end **search query classification** system to fetch and normalize **50,000 search queries**, applied **LLM prompt templates** via a Python script to categorize search queries into distinct categories, and stored search queries along with their classifications in **Amazon S3**.
- Fetched search queries and their classifications from Amazon S3, cached them in memory, and executed category-specific queries for precise, classification-aligned results.
- Built a prefix-based **search-query autocomplete system**: generated autocomplete suggestions from search-query prefixes, filtered noisy/misspelled terms with **LLM** prompts, and persisted the filtered search queries to Amazon S3.
- Loaded catalog titles and filtered search queries into a **Trie** and ranked autocomplete suggestions by a **weighted relevance score** combining popularity and search-query frequency, serving higher-scoring suggestions first.

Data Science Intern — January 2024 – April 2024
Innomatics Research Labs — *Hyderabad, India*

- Engineered a system to analyze and classify over **8,500 product reviews**, leveraging **Prefect** for ETL pipeline automation and scheduling.
- Trained sentiment analysis models using **BoW, TF-IDF, Word2Vec, and BERT**, achieving an **F1-Score of 0.92**, while utilizing **MLflow** for model management and experiment tracking.
- Developed and deployed a sentiment analysis web application on **AWS**, enabling real-time customer feedback insights.
- Constructed a **Generative AI Code Review application** using the **OpenAI API**, reducing code review time by **40%** with accurate bug detection and optimization suggestions.

Projects

Distributed Food Delivery Platform — January 2025 – March 2025

- Built a **full-stack food delivery system** with a **ReactJS** frontend and **Node.js/Express** backend using **MongoDB**, featuring secure authentication, customer/restaurant dashboards, order management, and REST APIs documented in **Postman**.
- Enhanced scalability by containerizing services with **Docker**, orchestrating with **Kubernetes**, integrating **Kafka** for asynchronous order processing, deploying on **AWS**, and managing application state with **Redux**; validated performance with **JMeter**.

Intelligent Transportation System — March 2025 – May 2025

- Developed a **3-tier distributed system** with modules for drivers, customers, billing, admin, and rides, using **Node.js**, **Express**, **REST APIs**, and **MongoDB**, and deployed via **Docker**.
- Implemented a **dynamic pricing algorithm** using the Uber Fares Kaggle dataset with machine learning models to balance supply–demand, and validated scalability with **Apache JMeter** load testing on **10k+** drivers, customers, and billing records.

SkinShade AI — March 2025 – May 2025

- Built a skin-tone classification system using **MTCNN**, **HSV**-based segmentation, and **K-Means clustering** to categorize tones into Light/Medium/Dark, leveraging the **CelebA** dataset.
- Deployed with **TensorFlow Lite** and a **Gradio UI**; integrated **OpenCV** and **PyTorch MTCNN** for detection and generated personalized **5-color** palettes to promote fairness in vision models.