

DALTON PHAM

☎ (+84)931-777-864

✉ daltonpham.we@gmail.com

🌐 [Linkedin](#)

🔗 [Leetcode](#)

🏠 [Github](#)

Summary

Data Science student with a **GPA of 3.89/4** and a strong foundation in **machine learning**, and **software engineering**. Proven ability to quickly adapt and implement complex solutions through internships at startups like **MarkovAI** and **Headstarter**, where I reduced load times by **40%** and built AI-driven solutions from the ground up. Proficient in **Python**, **C++**, and cloud platforms like **AWS**, with a demonstrated capability to rapidly learn new technologies and contribute effectively to projects.

Education

University Of Science

Sep. 2022 - May 2026

Bachelor of Data Science - GPA: 3.89/4

Ho Chi Minh, Vietnam

- **Relevant Coursework:** Data Structures and Algorithms (C++), Object-oriented programming (C++), Data Analysis, Probability and Statistics, Introduction to Data Science, Pattern Recognition.

Technical Skills & Certifications

Skills: C++, Typescript, Python, Pytorch, Tensorflow, Matplotlib, Scikit-learn, MySQL, Node.js, React.js, Express.js, Docker, Bash, Git/GitHub, HTML5/CSS3.

Cloud Computing: AWS (CloudFormation, Lambda, S3, EC2, CloudFront, Route 53).

Certifications: 3rd Runner-Up, Heading for the Future Project: Data Analysis (South Region); IELTS 7.0.

Experience

Software Engineer Intern

February 2024 – September 2024

MarkovAI - Start up | Python, Next.js, Appwrite SDK, Docker

Remote - San Francisco, CA

- Implemented an **advanced file upload feature** by configuring and optimizing hooks using the **Appwrite SDK**, efficiently handling file errors with the database, resulting in a **30% reduction in upload errors**.
- **Reduced page load times by over 40%** by applying lazy loading techniques with **Next.js**, optimizing resource loading and improving performance.
- Developed a responsive mobile web version, enhancing **user experience across devices and mobile traffic by 35%**.

Software Engineering Fellow

July 2024 – September 2024

Headstarter | Python, Rust, Node.js, Next.js, EC2, Pinecone, Generative AI

Remote - New York, NY

- Built **AI customer ChatBot** from **0 to 1** with a **team of 4**, leveraging **React.js**, **Next.js**, **OpenAI**, and **AWS**.
- Contributed to the **Biome (14k stars)** open-source project with a **team of 4** by converting **GraphQL** files to **Abstract Syntax Trees**, detecting errors with **GraphAL-ESLint**, and writing **comprehensive Rust function tests** for syntax error detection.

Webacy Remote Extern

June 2024 – July 2024

Extern | Python

Remote - New York, NY

- Conducted detailed analysis and labeling of over **30 smart contract vulnerabilities** by inspecting vulnerable accounts.
- **Validated peers' data labels** through collaborative reviews, ensuring an accuracy and reliability rate of **95%** across the dataset.
- Performed frequency and correlation analysis on risk tags using **Python and hierarchical clustering**, identifying the **top 10 most prevalent vulnerabilities**.

Projects

Flood Water Segmentation Using U-Net | Python, TensorFlow, Matplotlib

[GitHub](#)

- Developed a U-Net model for segmenting water regions in flood-affected areas using a dataset of **274 annotated images**.
- Implemented data augmentation techniques (random flips, rotations, zooms) to improve the **model's generalization**.
- Achieved a **validation accuracy of 79.12%** and validation loss of 0.3967 with a combination of binary cross-entropy and dice loss.
- Visualized segmentation maps and training curves to **analyze model performance** over time.

Clustering and Dimensionality Reduction | Python, Scikit-learn, PyTorch

[GitHub](#)

- Analyzed the **Vehicle Silhouettes dataset** using PCA, Kernel PCA, and autoencoders, **capturing up to 79.72% variance**.
- Developed a PCA-inspired Autoencoder with orthogonality constraints, **achieving a 57.30% explained variance ratio**.
- Applied the EM algorithm with various initializations (K-Means, Random, Hierarchical) and achieved **ARI scores up to 0.86**.

COVID-19 Sentiment Analyzer | Python, Scikit-learn, Matplotlib

[GitHub](#)

- Collected and analyzed **over 10,000 tweets** related to COVID-19 to gauge public sentiment.
- Implemented data preprocessing techniques to clean and prepare the dataset, **reducing noise by 25%**.
- Developed sentiment analysis models (Logistic Regression, Naive Bayes, SVM) achieving an **accuracy of up to 86%**.
- Visualized results using Matplotlib, including sentiment distribution over time and word clouds, providing **insights into public opinion trends**.