

Maxim Dokukin

Able to take in a lot of chaos and turn it into something manageable

maxdokukin@icloud.com | maxdokukin.com | github.com/maxdokukin | linkedin.com/in/maxdokukin

Skills

Edge ML: TensorFlow Lite, TFLite models, On-device inference, Real-time latency, TinyML, Sliding window, Multi model

Embedded Systems: C++, Embedded C, Keil uVision, MCU targets, I2S, I2C, DMA, Memory maps, Interrupt tuning

Model Ops: Quantization, Int8 quantization, Tensor arena, Model versioning, CI/CD, GitHub Actions, Performance profiling, Confusion matrices

Data Pipeline: Data scraping, Labeling, Augmentation, MFCC features, Preprocessing, Nightly jobs, Provenance index, Run books

Education

Masters of Science, Artificial Intelligence, San Jose State University May 2027

Bachelor of Science, Data Science, San Jose State University May 2025

- Summa Cum Laude, Presidents Scholar, AS Leadership Scholar
-

Experience

AI ML Engineer, Nuvoton (San Jose, CA) Aug 2024 - Present

- Productionized MCU audio classifier demo, achieved target **Latency, Memory Optimization, Embedded Systems**
- Built scalable scraping, labeling, and loader pipelines; enforced provenance and **Data Engineering, TensorFlow, Python**
- Implemented model ops, export metadata for edge, and **Model Optimization, Quantization, TensorFlow Lite, Keras**
- Refactored firmware with clear C/C++ boundaries; optimized **DMA, I2S, I2C, Memory Optimization**
- Established CI/CD, testing harness, runbooks, and cross-team reviews; emphasized **CI/CD, Documentation, Performance Testing, Communication**

ML Engineer Intern, Nuvoton (San Jose, CA) May 2024 - Aug 2024

- Boosted on-device sound classification 49%90%+ on MCU via **TensorFlow Lite, quantization, edge devices, TinyML**
- Designed YAML-driven reproducible pipeline for augmentation, balancing, preprocessing using **Python, data pipelines, preprocessing, augmentation**
- Trained 30+ lightweight models; exported minimal-operator **TensorFlow Lite** builds validated for **production-quality systems, testing**
- Engineered **MCU C++** inference harness, sized tensor_arena, prevented OOMs via **C++, performance profiling, on-device resource optimization**
- Automated nightly CI jobs to trainconvertflashtest; enabled **CI/CD, performance benchmarking, documentation, cross-functional collaboration, A/B testing**

Machine Learning Intern, Yandex (Remote) May 2023 - Aug 2023

- Preprocessed 1M+ records daily using **Python, Pandas, NumPy**, ensuring **Data Preprocessing** and **Data Cleaning**
- Automated 320GB/day remote processing with **Linux, Automation**, and robust **Data Pipelines**
- Drafted standardized preprocessing, led **Cross-functional collaboration** workshop, unified pipelines with **Git, Documentation, CI/CD**
- Visualized dataset biases with **Data Visualization** and **Data Analysis**, rebalanced training for improved **Machine Learning**
- Delivered production-ready preprocessing pipelines in Agile sprints, enabling reliable **NLP** and **Production-quality systems**

On-Campus Involvement

President, AI & ML Club @ SJSU

Jan 2025 - Present

Officer

Dec 2023 - Jan 2025

- Increased club attendance from 25 to 85 members
 - Expanded leadership team from 5 to 27 Officers
 - Collaborated with 2 industry leaders and organized on-campus speaker-events for them
 - Lead club projects with **reinforcement learning**, **neural networks**, **classification**, **clustering**, **regression models**, and **model optimization** using **TensorFlow** and **PyTorch**
-

Projects

Audio Data - Training

- Developed CNN audio classifier in **TensorFlow**, trained on MFCC spectrograms achieving 85% accuracy
- Optimized and **quantized** model, converted to **TensorFlow Lite** and auto-generated **C++** deployable
- Implemented **real-time** microphone pipeline with sliding windows for low-latency **on-device** inference
- Built robust **data pipelines**: spectrogram/MFCC preprocessing, augmentation, and validation for edge training
- Delivered end-to-end production-ready artifact with **model optimization**, **Python** tooling, **version control**, and **team collaboration**

Audio Data - Deployment

- Designed and **deployed ML models targeted for edge devices** using **TensorFlow Lite** for **production deployment**
- Implemented low-level audio I2S/I2C PDMA integration showing **SoC-specific constraints** and **ARM CPUs** familiarity
- Optimized inference with a custom averaging filter, improving accuracy — **model optimization techniques**
- Built edge runtime in **C++** with **embedded C** tooling (Keil), focusing on **on-device resource optimization**
- Led end-to-end deployment collaborating with team; strong **software engineering**, **testing**, **debugging**, and **collaboration**

Audio Data - Preprocessing

- Built 5-phase audio **Data Pipelines** in **Python**, scaling dataset from 2k to 90k
- Implemented silence, volume, and duration **Data Preprocessing** ensuring uniform one-second inputs
- Applied RIR, noise, time/volume/pitch **Data Augmentation** to improve model robustness
- Engineered reproducible pipeline with **Software Engineering**, **Debugging & Testing**, **Version Control (Github)**
- Prepared balanced, normalized dataset using **Data Cleaning**, **Data Normalization**, **Data Balancing** and analysis