Karan Sharma

krn-sharma.github.io | krn-sharma | Karan Sharma | krnsharma0209@gmail.com
+49 176 66236010

Summary

AI/ML engineer focused on MLOps and applied deep learning. Experience in finetuning and researching architecture of LLMs. Built reproducible training/inference pipelines on HPC with ZenML/MLflow, deployed CV/NLP models. Interested in AI applications and production-grade ML systems.

Education

•Georg-August-Universität Göttingen *Oct 2022 – Present*
M.Sc., Applied Computer Science Grade: 2.1
Thesis(Ongoing): Indian Sign Language Translation using multi modal Vision LLM architecture

•Indian Institute of Technology (IIT) Dharwad, Karnataka *Aug 2018 – Jul 2022*
B.Tech., Computer Science and Engineering Grade: 8.88/10 (German equivalent: 1.5)

Skills and Interests

Programming Languages: Python, C++, SQL
Frameworks & Libraries: PyTorch, TensorFlow, Django, Pandas, NumPy, OpenCV, Matplotlib, ZenML, MLflow
Domains: LLM, MLOps, Backend, Computer Vision, NLP, Data Analysis, REST APIs, JSON, XML, HPC, Knowledge Graphs, CI/CD, SLAM
Databases & Tools: MySQL, PostgreSQL, Git, Docker, Heroku, Linux, Anaconda, Napari, VS Code, Jupyter Notebook, Label Studio, Neo4j, Slurm

Experience

•Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG) *Jun 2024 – Present*
Working Student — MLOps On-site
– Engineered reproducible ML pipelines with ZenML and MLflow for experiment tracking, versioning, and consistent outputs across HPC environments.
– Achieved 85.58% segmentation accuracy and $\tilde{2}07.35$ TFLOPS using a 3D U-Net on BraTS MRI data within a ZenML-managed workflow.
– Built a transformer-based model for arrhythmia detection to improve ECG classification accuracy and enable real-time health monitoring applications.
– Created and used Slurm job scripts to submit/manage workloads on HPC, ensuring efficient resource utilization and parallelization.
– Integrated Jacamar CI with GitLab runners for secure, user-isolated Slurm submissions, streamlining CI/CD for ML pipelines on HPC.

•Histomography, Göttingen *Aug 2023 – Jan 2024*
Working Student — Image Processing On-site
– Implemented advanced image-processing algorithms for 3D X-ray tissue images; optimized big-data workflows and visualization using Napari.
– Applied blob detection and segmentation methods; engineered algorithms via grayscale histogram analysis to characterize tissue.
– Leveraged OpenCV, scikit-image, and SciPy to process large-scale 3D datasets robustly.

•LG Soft., Bangalore *May 2022 – Oct 2022*
Software Engineer On-site
– Developed a real-time Yoga Pose Correction feature for LG TVs using Keras (PoC) and C++ (production); integrated OpenPose and angle heuristics for posture feedback.
– Fixed critical WebOS media playback issues; optimized DASH streaming with GStreamer and automated testing via Python Selenium.
– Set up the first remote SoC board testing infrastructure with Device Farm, scaling to manage 100 boards for embedded validation.
– Mentored 2 interns and 3 new employees, improving code quality and understanding of WebOS architecture.

•LG Soft., Bangalore *Aug 2021 – Apr 2022*
Machine Learning Intern On-site
– Automated HbbTV testing using video analytics and computer vision for robust validation of broadcast standards.
– Initialized pose-correction research: built datasets (web scraping) and trained a pose-classification model achieving 90% accuracy.

Projects

•**Lego Mind Storm Robot** *Academics*

Autonomous robot for collection/delivery of wooden block with ArUco-based localization, homogeneous transforms, and SLAM
Demo video | Code

– Converted camera-frame coordinates to robot/world frames via homogeneous transformation matrices.
– Integrated SLAM for mapping and PID for motion control; implemented A* path planning on a discretized world map.

•**Disease Detection in Maize Crop** *Academics*

Sliding-window CNN object detection for leaf disease and nutrient deficiency

– Collected a custom field dataset and combined with publicly available datasets; achieved 98% detection accuracy for 5 different classes.
– Deployed to embedded devices with ArmNN (Linaro) on Odroid XU4 and TensorRT on Jetson Nano.

•**Active Learning for Text Labelling** *Academics*

Teacher–student self-supervised setup integrated with Label Studio

– Automated iterative labeling using a teacher–student pipeline to accelerate dataset curation.

•**QR-based Attendance System** *Academics*

Python Django web app with multi-user roles; REST API to Android app; PostgreSQL on Heroku

– Professors generate per-class QR codes; students scan via Android app to mark attendance.
– Designed REST interfaces and deployed PostgreSQL with RDBMS best practices.

Languages

•English - C1
•German - A2

Certifications

•Deep Learning Specialization *deeplearning.ai*