Yunus Emre KORKMAZ

+90-5310312626
official.yunusemrekorkmaz@gmail.com
github.com/dolphinium
linkedin.com/in/yunus-emre-k0rkmaz/

EDUCATION

Eskisehir Technical University *2019–2024*
B. Sc. in Computer Engineering, GPA: 3.37/4 *Eskişehir, Türkiye*

Eyüp Aygar Science High School *2014–2018*
High School Diploma, GPA: 91.06/100 *Mersin, Türkiye*

EXPERIENCE

Bewell Technology *July 2024 – Aug 2024*
Computer Engineer Intern as Artificial Intelligence Engineer *Eskişehir, Türkiye*

- Trained an Object Detection Model using YOLO, achieving 83% mAP@0.5 in detecting damaged buildings and extracting geolocation from drone imagery.
- Conducted performance benchmarks on YOLO models (V5-V8-V10) using Colab with GPU, optimizing for speed and accuracy trade-offs.
- Monitored and compared model performance metrics using CometML to identify the most effective architecture.
- Hosted an end-to-end demo website on Hugging Face Spaces, providing users with real-time access to building detection outputs.
- Managed code versioning and collaborated with teammates using GitLab, ensuring smooth project workflow and delivery.
- **Live Demo on Hugging Face Spaces:** huggingface.co/spaces/dolphinium/rescuenet-damaged-building-detection.
- **GitHub Repository:** github.com/dolphinium/rescuenet-damaged-building-detection.

Anadolu University Computer Research and Application Center *September 2023 – October 2023*
Computer Engineer Intern as Software Architect *Eskişehir, Türkiye*

- Developed and designed an end-to-end Web based Survey Application on Web Platform for Anadolu University by using .NET Framework and Angular.
- Utilized PostgreSQL for the database.
- GitHub and Microsoft Azure used for version control and task scheduling.

Hergele Mobility *March 2022 – October 2023*
Part-time Back-end Developer *İstanbul, Türkiye*

- Developed Web based Admin Dashboard for Electrical Scooters by using .NET Framework with MVC pattern.
- Utilized MongoDB for the database.
- Used Jira and GitHub for task scheduling and version control.

TECHNICAL SKILLS AND INTERESTS

- **Programming Languages:** Python (Proficient), C#, Java, Javascript
- **AI/ML Concepts:** Deep Learning, Machine Learning, Computer Vision, Object Detection (YOLOv5/v8/v10), Generative AI (LLMs, Image Gen - FLUX), Model Fine-tuning, RAG, NLP, Multimodal AI, OCR, Speech Processing (VAD, STT), Prompt Engineering, Agentic AI, Vector Search

- **AI/ML Libraries:** PyTorch, Hugging Face (Diffusers, Transformers, Datasets, Hub, Spaces), Scikit-learn, Pandas, NumPy, OpenCV, EasyOCR, FAISS, SpeechRecognition, TorchAudio, Gradio, LangChain (Basic)
- **Web Backend & APIs:** FastAPI, Celery, Flask, RESTful APIs, SQLAlchemy, Pydantic
- **Databases & Storage:** MongoDB, SQLite, Vector Databases (FAISS, Chroma)
- **Tools & Platforms:** Docker, Git/GitHub, Google Cloud , Azure (Basic), Amazon(Intermediate), CometML, Google Colab, Jupyter Notebooks, Linux/macOS CLI

Field of Interest: Applying AI/ML to Real-World Problems, Computer Vision, Generative AI (Model Fine-tuning, Image/Text Generation), NLP (RAG, Document Intelligence, Speech Analysis), Multimodal AI Systems, Building Production-Ready AI Systems, MLOps.

PROJECTS

Fine-tuning FLUX.1-dev Using PEFT (LoRA) March 2025

Fine-tuned the Flux.1 Dev text-to-image model using a curated dataset and advanced techniques to generate artwork specifically in the Impressionist style.

- Fine-tuned the Flux.1 Dev diffusion model using PEFT (LoRA) techniques on NVIDIA A100 GPUs to specialize in generating Impressionist-style images.
- Curated and published a high-quality dataset of 1,000 Impressionist paintings from WikiArt, ensuring balanced genre distribution and applying quality filters.
- Engineered a robust, automated image captioning pipeline leveraging the Google Gemini API with batch processing, checkpointing, and error handling to enrich dataset quality.
- Deployed the fine-tuned LoRA model weights to Hugging Face Hub, enabling public accessibility and integration via the Diffusers library.
- Implemented comprehensive training visualizations (progress grids, step evolution) and performance tracking for effective model development analysis.
- Managed the end-to-end project lifecycle, including data preparation, model training, custom pipeline development, deployment, and documentation using Python, PyTorch, and the Hugging Face ecosystem.
- **GitHub Repository:** github.com/dolphinium/flux-impressionism-finetuning.
- **Hugging Face Model:** dolphinium/FLUX.1-dev-wikiart-impressionism
- **Hugging Face Dataset:** dolphinium/wikiart-impressionism-curated
- **Tools and technologies used**: Python, PyTorch, Hugging Face (Diffusers, Datasets, Hub, Spaces, PEFT), Flux.1 Dev Model, Google Gemini API, AI-Toolkit, NVIDIA A100 GPUs, Google Colab Pro, Git, GitHub, Jupyter Notebooks, Pandas, NumPy, Pillow, OpenCV, aiohttp, asyncio, Tensorboard, Matplotlib, Seaborn, Gradio.

Automated Call Classification System February 2025

Developed a production-grade system for automated analysis and classification of customer service call recordings, achieving ***over 95% accuracy.***

- Designed and implemented a scalable, microservices-based architecture using FastAPI, Celery, MongoDB, and Redis for high-throughput audio analysis.
- Integrated Google's Gemini model and Silero VAD for intelligent call classification and precise voice activity detection, respectively.
- Developed robust error handling and retry mechanisms to ensure production reliability, processing thousands of audio files daily.
- Achieved **>95%** classification accuracy in production with real customer data, optimizing customer service operations through intelligent call categorization.
- **GitHub Repository**: github.com/dolphinium/call-classification-system
- **Tools and technologies used**: Python, FastAPI, Celery, MongoDB, Redis, Docker, Gemini, Silero VAD, PyTorch, TorchAudio, SpeechRecognition, FFmpeg

LANGUAGES

Turkish: Native
English: Fluent
German: Beginner