Gül Oymak

Zurich, Switzerland | goymak@ethz.ch | +41 78 402 91 64 |
linkedin.com/in/gul-o-87009a132 | github.com/guloyy

Education

ETH Zurich, MSc in Computer Science — September 2024 - Ongoing
- **Major:** Machine Intelligence

ETH Zurich, BSc in Computer Science — September 2020 - May 2024
- GPA: 4.79/6.0

Istanbuler Gymnasium, High School Diploma — September 2015 - June 2020
- **Valedictorian** with Abitur grade 1.2

Coursework

Probabilistic AI, Machine Perception, Large Scale AI Engineering, Interactive Machine Learning, Large Language Models, Information Security Lab, Big Data, Advanced Systems Lab

Experience

Software Engineer, Intern, Credit Exchange AG — January 2024 – September 2024
- Transferred legacy database to an Azure PostgreSQL Database to ensure improved data management and accessibility.
- Implemented a secure and reliable cookie capturing algorithm to enable automatic login and testing on the company platform, streamlining the testing process.
- Developed an algorithm to automatically identify and resolve inconsistencies between company data and partner data for increased data accuracy and reliability.

Researcher for Bachelor Work, Paul Scherrer Institute, Swiss Data Science Center — September 2023 – March 2024
- Collaborated with the Swiss Data Science Center and Paul Scherrer Institute on a project that utilizes in-situ sulfate observations in Europe and the Comprehensive Air Quality Model data within a Random Forest Model, aiming to enhance the accuracy of model predictions.
- The primary objective is to use machine learning to generate high-resolution sulfate maps and identify the sources, temporal patterns, and possible impacts of sulfate aerosols.

Teaching Assistant for Visual Computing, ETH Zurich — September 2023 – December 2023
- Conducted weekly lab sessions for students in Python and OpenGL for the course Visual Computing.
- Created as well as helped with the maintainance of written and applied course materials that emphasize real-world applications.

Project Tutor at Summer School, Freie Universitaet Berlin — July 2019 – Aug 2019
- Taught introductory Python and Arduino to new students at the summer school and supervise their projects after completing the computer science summer school for high-school students with a Stellar Performance.

Projects and Papers

Post-Training Quantization of State Space Models: A Practical Study — April 2024 - July 2024
- In this project, we compared the memory-accuracy tradeoff between state-of-the-art transformer models and structured state-space models (SSMs) with fewer than 7 billion parameters. We applied post-training quantization in both moderate (8-bit) and extreme low-precision (4-bit) scenarios, using simple

Round-to-Nearest and GPTQ quantization schemes under the guidance of Carmen Amo Alonso. Our analysis revealed that SSMs outperform transformer models with an equivalent number of parameters at 8-bit precision. However, in the 4-bit precision setting, smaller transformer models (<1 billion parameters) demonstrated higher accuracy than SSMs. Despite this, for models with at least 1 billion parameters, 4-bit SSMs consistently outperformed transformer models of comparable sizes.

- Tools Used: Python, Cuda

Interactive Chain-of-Thought Visualization System February 2025 – May 2025

- Built an interactive system to visualize chain-of-thought (CoT) reasoning by showing step-wise answer distributions derived from secondary LLM prompts and vocabulary logits.
- Represented model confidence using entropy and tracked correctness at each reasoning step.
- Enabled what-if analysis by allowing users to overwrite specific steps and observe whether the model recovers or changes its final answer.
- Supported visualization of global patterns to assess when the model is likely to be correct or incorrect.
- Focused on mid-tier LLMs to capture a wide spectrum of correct and flawed reasoning paths.

Accelerating Mathavan's Billiard Ball Collision Algorithm February 2025 – June 2025

- Optimized a physics-based billiard collision simulation written in C, achieving a $1.93\times$ runtime speedup through scalar and algorithmic improvements.
- Contributed float-based and vectorized implementations with trade-offs in numerical precision; ensured accuracy within 1% relative error.
- Tools Used: C, Clang, Linux PMCs, Python (for benchmarking and plotting).

Skills

Languages: Python, SQL, C++, Java, TypeScript, C

Software: Docker, Microsoft SQL Server, FastAPI, Azure Databases, Blender, OpenGL, Node.js, Apache Spark, AWS

Machine Learning Frameworks: TensorFlow, Scikit-learn, PyTorch, OpenCV

Language

English: Advanced
German: Advanced
Turkish: Native