PETAR PETROV

+31 6 53873261 ✉ petizad55@gmail.com Petar Petrov petaripetrov Personal site

EDUCATION

MS Computer Science Feb. 2024 - Expected 2026
Technische Universiteit Delft

MS Computer Science Sep. 2023 - Feb. 2024
Vrije Universiteit Amsterdam & Universiteit van Amsterdam

BSc (Hons.) Computer Science Sep. 2020 - Aug. 2023
Vrije Universiteit Amsterdam
- GPA: 8.0

EXPERIENCE

Teaching Assistant Sep. 2024 - Oct. 2024
Technische Universiteit Delft *Delft, The Netherlands*
Machine & Deep Learning: Answering questions about the course content, focusing on Statistical Machine Learning and Deep Learning;

Junior Software Engineer Sep. 2020 - March 2024
Amexis *Plovdiv, Bulgaria*
My job includes participating in Scrum, working on new features, and fixing issues with existing code, such as bugs, inefficiencies, and inconsistent coding style. The work is split between the frontend and backend on multiple projects ranging from user management platforms to large-scale applications.

Teaching Assistant Jan 2023 - May 2023
Vrije Universiteit Amsterdam *Amsterdam, The Netherlands*
Web Technology: Helping students with assignments, explaining web concepts, and resolving group conflicts;

Software Design: Managing student groups through a project. Primarily helping them pick out appropriate features and requirements, and making sure that cooperation works smoothly;

Computer Networks: Guiding students through assignments and checking homework.

SKILLS

Technical Skills	.NET, Angular, TypeScript, JavaScript, C/C++, Rust, Python, SQL, PostgreSQL, Scrum
Soft Skills	Leadership, Adaptability, Work ethic, Project Management, Problem-Solving Eagerness to learn, Teamwork
Languaages	English (native), Bulgarian (native), Dutch (limited)

PROJECTS

Arenas in Rust: encoding and benefits Bachelor thesis looking into cyclic types in Rust. My work involved condensing Rust documentation to explain the limitations around cyclic types and testing Arenas as a framework for self-referential data structures in Rust and C++. My findings, implementation, and a condensed explanation of Rust's type system can be found in a paper summarizing the research.

Reproducibility of "EG3D: Efficient Geometry-aware 3D Generative Adversarial Networks" Project aimed at recreating the results of "EG3D: Efficient Geometry-aware 3D Generative Adversarial Networks". My work primarily focused on getting the model to work on Kaggle and extracting a feature vector which allows us to shift the gender of generated subjects.

Retrieval Augmented Generation (RAG) for Open Domain Complex Question Answering In this research project, we have conducted ex-periments to evaluate what kind of context is valuable to provide the LLM with when answer-ing compositional questions.