Cătălin-Alexandru Rîpanu

Bucharest, Romania

☎ (+40) 771 067 932 ✉ catalin.ripanu@upb.ro Linkedin profile Github profile

Relevant Education

POLYTECHNIC University of Bucharest — September 2024 – July 2026

Faculty of Automatic Control and Computers — *Bucharest, Romania*

- Pursuing a **Master of Science Degree** in **Artificial Intelligence**, with all classes conducted in English.
- Relevant **coursework:** Deep Neural Networks, Computer Vision, Knowledge Representation & Reasoning, Type Systems & Functional Programming, Multi-Agent Systems, Natural Language Processing, and Symbolic & Statistical Learning

POLYTECHNIC University of Bucharest — September 2020 – July 2024

Faculty of Automatic Control and Computers — *Bucharest, Romania*

- Earned a **Bachelor's Degree** in Computer Science and **Engineering**, achieving a **GPA** of 9.805 / 10.00.
- Relevant **coursework**: Artificial Intelligence, Machine Learning, Quantum Computing, Algorithms Analysis & Design, Compilers, Data Structures, Numerical Methods, Formal Languages & Automata Theory, and Programming Paradigms

Relevant Work Experience

POLYTECHNIC University of Bucharest — September 2024 – Present

Teaching Collaborator and Associate Researcher of AI-MAS Laboratory — *Bucharest, Romania*

- Focused on developing and implementing diverse Deep Learning neural architectures across Computer Vision, Natural Language Processing, and various AI Learning Methodologies, including Knowledge Distillation and Federated Learning

POLYTECHNIC University of Bucharest — February 2022 – Present

University Graduate Teaching Assistant — *Bucharest, Romania*

- Taught students Data Structures & Algorithms, Programming Paradigms, Numerical Methods, and Machine Learning.
- Assisted in grading student projects and served as an invigilator alongside professors during midterms and final exams.

Personal Projects

Neural ODE Generative Model with Quantum Vision Transformers | *TensorCircuit for Quantum, Jax* — **July 2024**

- Implemented using **Jax** & **Flax** a *Generative neural network using Quantum* tested on **CIFAR10** and **IMDB** samples.
- Designed a Variational Quantum Circuit in **TensorCircuit**, harnessing **Quantum Entanglement** through Bell states.
- Created a Quantum Vision Transformer Architecture that leverages Runge-Kutta Numerical Methods for better scores.
- Evaluated and compared it alongside a model presented at **NeurIPS 2021**, showing promising results in Quantum AI.

IoT Platform using Microservices for Time Series Data | *MQTT, Grafana, Portainer, CI/CD, Flask* — **June 2024**

- Implemented a *Platform* for manipulating Numerical Data coming from a large number of Internet of Things devices.
- Deployed **Grafana** in a Docker environment to visualize data and gain analytical insights through edited dashboards.
- Utilized **Portainer** in Docker Swarm to **monitor** Load Balancing effects of container **replicas** using **multiple** nodes.
- Employed **GitLab's CI/CD** for further comprehending builds & tests automation and software development practices.

COOL Compiler with ANTLR v4.0 Generator | *Lexer, Parser, Code Generation, Java, MIPS, COOL* — **Feb 2024**

- Developed a **Java-based** *Compiler* for an Object-Oriented Programming language, incorporating **basic inheritance**.
- Designed **Lexical Analysis** utilizing **ANTLR4.13** to construct a grammar that accurately recognizes language tokens.
- Defined Resolution and Definition Pass traversals using **Visitor Pattern** for creating Syntactic and Semantic Analyzes.
- Developed Code Generation for translating any COOL code into MIPS Assembly. Used the *SPIM* Simulator for testing.

2016 Halite Bot | *Algorithm Design and Analysis, C++, Machine Learning, Artificial Intelligence* — **May 2022**

- Implemented in **C++** a *Halite bot* using a **Runtime Engine** integrated within a **Framework** given by the organizers.
- Processed in a **Greedy** way the cells with the highest scores first to let the bot conserve its strength score in the match.
- Developed a **logic** such that if a border cell cannot attack, it will look for a neighboring cell with which it can combine.
- Implemented a **surplus strength** redistribution algorithm that evenly allocates excess power score to neighboring cells.

Awards

National Student Mathematics Competition "Traian Lalescu" — November 2021

2^{nd} *Year Contestant* — *Transilvania University of Brasov, Romania*

- Earned honorable mention in the **Complex Analysis** section at the **National phase** of the Mathematical olympiad.

Skills

Technical Skills

- Intermediate Knowledge: Data Structures, Algorithms, C/C++, Python, Java, Networking, Numpy, Pandas, Pytorch
- Basic Knowledge: TensorFlow, Jax / Flax, TensorFlow Quantum, DevOps, CUDA, Flask, SQL, Haskell, Prolog, *React*

Languages

- Romanian: Native Speaker
- English: Professional Level
- French: Good Command