Alexander Buyantuev

+7 911 292 71 53 | alexbuyan.dev@gmail.com | alexbuyan | alexbuyan

Education

HSE University Saint-Petersburg, Russia

Bachelor of Applied Mathematics and Computer Science *Sep. 2020 - Aug. 2024*

- **Relevant courses**: *Algorithms and Data Structures, C/C++, Java, Python for Web, Python for Backend, Machine Learning, Databases, Computer Networks, Calculus, Linear Algebra, Discrete Mathematics, Probability Theory, Math Statistics*
- **Yandex Data School courses**: *Machine Learning, Natural Language Processing, Reinforcement Learning*
- **AI Talent Hub courses**: *Project Management in Data Science*

Skills

Programming languages: Python, C/C++, Java
Technologies and Frameworks: git, SQL, PyTorch, Docker, FastAPI
Languages: Russian, English (C1)

Work Experience

RL Researcher *Sep. 2023 - Present*

Huawei R&D, Network Scheduling Team Saint Petersburg, Russia

Linear erasure codes construction using reinforcement learning | *Python* Sep. 2023 - Present

- Researched the application of **DDQN** and **SAC** algorithms for designing linear erasure codes
- Developed an evaluation strategy using FastAPI to **speed up** the training up to **50 times**
- **Increased the accuracy** of evaluation from $\mathbf{10^{-3}}$ to $\mathbf{10^{-6}}$
- **Compared** obtained linear erasure codes with FlexFEC and Reed-Solomon codes in Bernoulli and Simple-Gilbert loss models
- **Showcased the advantage** of RL designed codes in **performance** and **decoding complexity**

Software Engineer Intern *Nov. 2022 - Present*

Huawei R&D, Cangjie Team Saint-Petersburg, Russia

CSV support for Data-Driven Testing in Cangjie | *Cangjie* Sep. 2023

- Implemented CsvParser in Cangjie to **parse** data from CSV files
- Developed CsvStrategy to provide data for unit tests and **contributed** it to Cangjie Test Framework

LLVM IR decompiler for Cangjie | *C++, Python, GoogleTest* Nov. 2022 - June 2023

- Designed a tool to represent LLVM IR module in C-like format that restores packages, classes and functions from Cangjie source code to **speed up** compiler's generated code analysis
- Implemented LLVM GEP instruction printer to show class field and it's type when accessed by the pointer to **improve** code readability
- Downloaded source code from 300+ open projects on Cangjie and created test cases from source code to **test** the tool
- Developed a parallel testing framework that runs 30 test cases with 100000 lines each under 1 minute to **fix bugs** in my tool
- **Distributed** the tool inside Cangjie Team for analysis of compiler's generated code by other developers

Projects

PDF Editor with LaTeX support | *Java* Mar. 2022 - June 2022

- Designed a converter of UI objects to PDF document to **transfer** project's data to PDF file
- Implemented rendering of LaTeX equations to **allow** users to work with math formulas
- Developed a utility to download and save user's files to **enable** users to save their projects
- Added font support in UI and PDF to **empower** the customization of documents

Messenger with Trello boards | *C++, PostgreSQL, Trello API* Jan. 2021 – May 2021

- Created database to **store** users' information
- Implemented curl library wrapper to work with Trello API to **support** Trello boards
- Developed server's functionality to **handle requests** to the database

Parser generators comparison | *Python, Java, ANTLR4, Parglare* Oct. 2021

- Researched basic functionality and limitations of ANTLR4 and Parglare to **compare** them with other parser generators
- Compared generators' performance on ambiguous grammar recognition to **collect** data for the report
- **Described** research results in the report