

教育背景

南京大学	2023.09 - 2026.06 (预计)
硕士, 计算机科学与技术   <a href="#">Pascal Lab</a> , 导师: 李██   研究方向为程序语言与程序分析	
北京航空航天大学	2019.09 - 2023.06
本科, 计算机科学与技术   GPA: 3.84/4.00	

工作经历

NVIDIA · GPU 编译器 LLVM 后端实习生	2025.02 - 至今
<a href="#">Rust Foundation Fellowship Program</a> · 开源社区资助	2024.09 - 2025.09

作为 rust-analyzer 的 maintainers 之一, 负责维护 rust-analyzer (Rust 语言官方 IDE) 等 rust 语言基础设施

- 社区贡献排名 **21/972**, 解决 **~70 issues**, 参与维护过绝大多数模块;
  - 参与解决 issues、会议讨论、PR 审核等社区维护工作; 为项目添加控制流导航、快照测试更新等多项功能;
  - 为 IDE 的 unicode 断字和断行模块编写了 NEON 下的 **SIMD** 实现, 使该模块在 ARM 平台上提速 **6.5 倍**;
  - v0.3.1992 **事故救火**: 社区在发布小版本 4 小时后, 发现该版本存在导致资源耗尽且无法结束进程的恶性 BUG。本人在 3 小时内定位到错误算法, 并设计新算法解决了问题。该紧急修复控制了事故影响范围, 避免影响全球 Rust 开发者的工作。

奖项荣誉

- 2022 年**国家奖学金** (该学年专业排名 1/195); 北京航空航天大学优秀毕业生;
- 2021 年全国大学生计算机系统能力大赛 · **编译系统设计赛** (华为毕昇杯) **一等奖**;
- 蓝桥杯 C++ 程序设计竞赛北京赛区一等奖、国赛三等奖;
- 另获其他各类省、校级奖项与奖学金十余次。

项目开发

<b>Vizsla</b> · 现代化芯片设计 IDE (Rust / SystemVerilog)	<a href="#">roife/vizsla</a> (WIP)
<ul style="list-style-type: none"><li>基于<b>增量计算架构</b>, 实现了面向 SystemVerilog 的语义分析框架以及 IDE 基础设施, 旨在为芯片设计配备现代 IDE 功能;</li><li>项目在功能、性能与可用性等指标上均达到<b>业界先进水平</b>: 已完成代码导航、重构、代码补全等<b>数十项</b>现代 IDE 特性, 并能够做到<b>毫秒级延迟</b>; 项目已通过语言服务器协议适配 VS Code、Emacs 等主流编辑器。</li></ul>	
<b>Ailurus</b> · 实验性质的编程语言及工具链设计探索 (Rust)	<a href="#">roife/ailurus</a> (WIP)
<ul style="list-style-type: none"><li>基于 <b>Martin-Löf 类型论</b>; 支持<b>双向类型检查</b>、<b>dependent type</b>、indexed inductive types、pattern matching、module system 等特性; 使用 NbE 进行等价检查, 实现了 propositional equality, 支持定理证明;</li><li>项目包含解释器、REPL 和一个 VM; VM 使用 SSA 作为 IR, 能够执行 GVN 等运行时优化, 并通过 JIT 生成目标代码。</li></ul>	
<b>LLVM-Lite</b> · 面向深度学习算子的轻量编译器 / 本科毕设 (C++ / LLVM-IR)	<a href="#">roife/llvm-lite</a>
<ul style="list-style-type: none"><li>项目包含一个轻量编译器和经裁减的 LLVM codegen 模块, 旨在利用端侧推理设备的形状信息对深度学习算子进行二次优化;</li><li>毕设获<b>优秀</b>评价; 成功将 2D 卷积算子和 softmax 算子的用户侧<b>推理时间降低 6%</b>, 并将生成的二进制目标文件减小 <b>38%</b>;</li></ul>	
<b>Ayame</b> · C 子集编译器 / 毕昇杯比赛项目, 合作 (Java / LLVM-IR / ARM)	<a href="#">No-SF-Work/ayame</a>
<ul style="list-style-type: none"><li>主要负责完成基于图着色的<b>迭代寄存器合并算法</b>, 以及<b>指令调度</b>、<b>窥孔优化</b>等 Machine IR 上的优化; 并用 docker 搭建了本地 CI;</li><li>项目在 SSA IR 与 Machine IR 上进行优化, 在比赛中获<b>一等奖</b>。本项目在比赛中总排名第二, 在<b>近一半样例上排名第一</b>, 并在 1/3 的样例上优化效果超越 gcc -O3 与 clang -O3。</li></ul>	

开源社区贡献

- 目前是 **Rust-lang Member** (rust-analyzer contributors team) 的一员; 在 rust 社区的工作集中在 [roife/rust-lang/rust-analyzer](#) 上, 同时也贡献过 [roife/rust-lang/rust](#), [roife/rust-lang/rust-clippy](#), [roife/rust-lang/rustup](#), [roife/rust-lang/rust-mode](#) ;
- [roife/llvm/llvm-project](#), [roife/clangd/vscode-clangd](#), [roife/google/autocxx](#), [roife/yuin/goldmark](#), 更多项目见 GitHub。

专业技能

编程语言	不局限于特定编程语言。熟悉 C, C++, Rust, Java, Python, JavaScript/TypeScript, Verilog/SystemVerilog; 也使用过 Ruby, Swift, OCaml, Haskell, Coq, Agda 等。
程序语言理论	了解形式语义、形式验证与计算理论的相关知识, 熟悉 Hindley-Milner 等 <b>类型系统</b> 的理论和实现。
编译器	<b>4 年经验</b> 。熟悉解释器/编译器的全 <b>pipeline</b> 开发; 熟悉多种 IR (如 SSA / LLVM-IR, DBI, ANF, MLIR 等) 和 <b>编译优化</b> (如 Mem2Reg, GVN/GCM, 寄存器分配, 指令调度等); 了解 LLVM, 熟悉部分优化和 codegen 模块; 了解 Mark-Sweep、Copying 等常见的 GC 算法原理和实现。

IDE 开发	2 年经验，熟悉基于增量计算的 IDE 架构，尤其熟悉 rust-analyzer 和 clangd 的架构和实现；了解 VS Code、Emacs 等各主流编辑器的插件开发，熟悉语言服务器协议（Language Server Protocol）的规范和实现。
程序分析	熟悉常见静态分析算法，如数据流分析、区间分析、IFDS 等，熟悉基于不同敏感性的指针分析算法；
系统编程	了解体系结构和操作系统相关知识，能进行汇编级的开发调试工作，了解 Docker, GDB, CMake 等工具。
开发环境	熟悉 Emacs；习惯在 macOS / Linux 下工作；能熟练使用 AI 提高工作效率。

其他

- 助教：程序设计基础（2020 秋），面向对象设计与构建（2021 秋，2022 春），编译原理（2024 春）。
- 外语：英语（CET-6）；