

教育经历

东北大学，软件工程专业，本科，中国

2021 年 9 月 – 2025 年 6 月

部分必修课程：操作系统, 计算机网络, 计算机体系结构

工作经历

腾讯科技有限公司 WXG 微信事业群

2024 年 12 月 – 至今

技术架构部，基座大模型预训练/后训练工程师

- 负责微信内部基座大模型预训练/后训练的框架开发工作，包括模型训练加速，显存优化等。
- 参与 DeepSeek Infra 的复现工作，使用 ThunderKittens 重构 DeepGEMM，负责框架内 80% 的 Kernel 开发工作。

捷并思研究院，远程


2020 年 1 月 – 2020 年 12 月

同伦类型论与依值类型，交互式定理证明器 Arend 开发实习

- 使用 gradle composite build 和 buildSrc 等复杂的构建功能改善编译速度、提高构建自动化程度。
- 实现了一些语言特性和 IDE 插件功能，如 section 语法、卫生宏、优化的 Fin 类型、语义高亮等。
- 开发了一个可扩展的 REPL 引擎，并提供命令行实现（可通过上下文补全，使用 jline3 框架）、IntelliJ IDEA 实现（可与打开的项目进行交互，支持补全、高亮和跳转）。
- 设计并实现了一个表达式类型检查的调试器，支持显示局部变量和表达式栈帧，同时支持单步调试。


项目经历

Aya Prover，实用的依值类型系统实现（职位：项目组长）

 [aya-prover/aya-dev](https://github.com/aya-prover/aya-dev)


- 支持依值类型、依值模式匹配及重叠情况一致性检查、高阶归纳类型、泛化代数数据类型（已发表文章）、分层宇宙、立方类型论特性和隐式参数的推导。
- 能导出繁饰结果到网页或 L^AT_EX。能将解释执行时的闭包 JIT 编译到 Java 字节码（使用 HOAS 接入原本的语法树）。能根据模式匹配覆盖情况智能对模式细化。同时支持 VSCode 的语言服务器协议和 IntelliJ 的代码分析框架。提供 jlink 二进制分发。

IntelliJ Pest，给 IntelliJ Platform 开发的 Pest 文法语言插件


 [pest-parser/intellij-pest](https://github.com/pest-parser/intellij-pest)

- 支持含语义信息的高亮、错误检查、定义跳转、变量补全、提取定义、内联定义以及与 Rust 插件集成。
- 提供实时高亮功能——可根据语法定义为用户代码动态提供高亮以测试语法定义文件，并支持导出 HTML。

VSCode extension for Arend，Arend 语言服务实现，基于 lsp4j 框架和 Arend 编译器内部接口。

 [ice1000/vscode-arend](https://github.com/ice1000/vscode-arend)

Arend IO，实验性 Arend IO 库，实现了 unsafePerformIO 和简单的 IO 操作。

 [ice1000/arend-io](https://github.com/ice1000/arend-io)

技能

- 编程语言：泛语言（编程不受特定语言限制），且尤其熟悉 Java Kotlin Rust C# Agda Haskell Arend，较为熟悉 Dart C C++ F# F★ Idris Perl MATLAB（排名均不分先后）。
- 编译器：理解局部无名表示法、惰性代换 (ES)、ANF、(P)HOAS 等技术，理解语义正规化 (NbE) 及其使用的惰性闭包技术。熟悉大部分 parser 生成器，理解布局的语法解析。
- Kotlin/Java：10 年开发经验，熟悉 JNI, JPMS, Gradle, Kotlin coroutines, Swing。
- 类型论：理解 Martin-Löf 类型论、余归纳法、同伦类型论、立方类型论；熟悉 Idris, Agda（5 年经验，编译器和标准库贡献者），Arend 和一些 Lean/F★/Coq。
- JetBrains MPS：理解 面向语言编程 的概念和应用。
- IDE 工具开发：6 年开发经验，熟悉 IntelliJ 平台的基础设施（开发了 [Julia](#)、[DTLC](#)、[Pest](#)、[Kala Inspections](#) 等插件），同时了解 VSCode 的插件开发。
- 移动端开发：2 年开发经验，熟悉 Flutter、Android。
- 开发工具：能适应任何常见编辑器/操作系统，有使用 YouTrack、Jira、GitHub、BitBucket、Slack、JetBrains Space 等团队协作工具的经验。

其它

- 相关个人页面链接（请使用支持超链接的 PDF 阅读器）：[Crates.io](#) 主页，[IntelliJ](#) 插件市场主页
- 语言：English - 熟练 (托福 100)，汉语 - 母语水平
- 开源贡献：<https://ice1000.org/opensource-contributions>，向 [agda](#), [Arend](#), [libgdx](#), [jacoco](#), [KaTeX](#), [shields.io](#), [grpc-rs](#), [intellij-solidity](#), [intellij-haskell](#), [intellij-rust](#), [TeXiFy-IDEA](#), [rust-analyzer](#) 等项目贡献过代码
- [StackOverflow](#): 6000+ 声誉，同时也在 [Proof Assistants](#)（5000+ 声誉）和其他 [StackExchange](#) 子站活跃
- 获取此简历的最新更新：中文版本 <https://tinyurl.com/ya4urea8>，英语版本 <https://tinyurl.com/y2v59t36>，单页版本（仅英语）：<https://tinyurl.com/y8xdlfug>
- 在 [CodeWars](#) 上，以 Haskell、Agda 和 Idris 为主，达到 1 dan，全站排名 #111（前 0.020%）

论文发表 & 预印本

- [1] T. Zhang, “A Simpler Encoding of Indexed Types,” in Proceedings of the 6th ACM SIGPLAN International Workshop on Type-Driven Development, in TyDe '21. Republic of Korea: ACM, 2021. doi: [10.1145/3471875.3472991](https://doi.org/10.1145/3471875.3472991).
- [2] T. Zhang, “Elegant elaboration with function invocation.” [Online]. Available: <https://arxiv.org/abs/2105.14840>
- [3] T. Zhang, “A tutorial on implementing De Morgan cubical type theory.” [Online]. Available: <https://arxiv.org/abs/2210.08232>
- [4] T. Zhang, “Three non-cubical applications of extension types.” [Online]. Available: <https://arxiv.org/abs/2311.05658>
- [5] T. Zhang, “Two tricks to trivialize higher-indexed families.” [Online]. Available: <https://arxiv.org/abs/2309.14187>