

# Fan Zhang

zfan38551@gmail.com ·  <https://github.com/zfan2356>

---

## Education

B.S. in Software Engineering at **Northeastern University**, NE, China

Sep, 2021 – Jun, 2025

Selected courses: Operating Systems, Computer Networks, Computer Architecture

---

## Work Experience

**Tencent**, WXG WeChat Technology Group

Dec, 2024 – Present

Technology Architecture Department, Pre-training/Post-training Engineer of Base Model

- Responsible for the framework development of pre-training/post-training of base model in WeChat, including model training acceleration and memory optimization.
- Participated in the reimplementation of DeepSeek Infra, using ThunderKittens to refactor DeepGEMM, and responsible for the development of **80%** of Kernels in the framework.

**JetBrains Research**, Remote

Jan, 2020 – Dec, 2020

HoTT and Dependent Types, Interactive Theorem Prover Development

- Improved the language/IDE, such as sections, hygiene macros, `Fin` type with elaborative subtyping, semantic highlighting, etc.
- Created a debugger for inspecting bidirectional type-checking and REPL in both CLI and IntelliJ IDEA.

---


## Related Projects

**Aya Prover**, Practical Implementation of Dependent Types (role: project leader)

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

- Supports dependent types, dependent pattern matching with confluence check for overlapping clauses, higher inductive types, GADTs (paper published), hierarchial universes, cubical type theory features, and implicit arguments.
- Can export elaboration result to HTML or  $\text{\LaTeX}$ . Can JIT-compile closures into JVM using HOAS, Can refine patterns using coverage information. Supports both LSP in VSCode and IntelliJ PSI. Provide jlink binary releases.

**IntelliJ Pest**, Pest language plugin for IntelliJ Platform

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

- Semantic-based highlighting, completion, navigation, definition extraction/inlining, and Rust plugin integration.
- Provides live preview – test grammar files by dynamically highlighting user code according to the grammar on the fly. These highlighted code could be exported to HTML.

---

## Skills

- Programming Languages: multilingual (not limited to any specific language), especially experienced in Java Kotlin Rust C# Agda Haskell Arend, comfortable with Dart C C++ F# F★ Idris Perl MATLAB (in random order).
- Compiler: understand techniques like locally nameless, explicit substitution, ANF, (P)HOAS (in LF & logic programming), and NBE.
- Kotlin/Java: **10 years of experience**, familiar with JNI, JPMS, Gradle, Kotlin coroutines, and Swing.
- Type Theory: understand Martin-Löf type theory, coinduction, HoTT, and Cubical, familiar with Idris, Agda (**5 years** of experience, contributor), Arend and some Lean/F★/Coq.
- IDE Tooling: **6 years of experience**, familiar with the IntelliJ Platform infrastructure (created [Julia](#), [DTLC](#), [Pest](#), [Kala Inspections](#), etc.), also have experience with VSCode plugin development.
- Tools: editor-agnostic, have experience with team tools like YouTrack, Jira, GitHub, BitBucket, Slack, JetBrains Space and more.

---

## Misc

- Profile links (please use a PDF reader with hyperlink support): [Crates.io](#), [IntelliJ Marketplace](#)
- Languages: English - fluent (TOEFL 100), Chinese - native speaker
- Open-source contributions: <https://ice1000.org/opensource-contributions>, contributed to **agda**, **Arend**, **libgdx**, **jacoco**, **KaTeX**, **shields.io**, **grpc-rs**, **intellij-solidity**, **intellij-haskell**, **intellij-rust**, **TeXiFy-IDEA**, **rust-analyzer** and other projects
- [StackOverflow](#): 6000+ reputation, also active on [Proof Assistants](#) (5000+ reputation) and [other StackExchange sites](#)
- Latest revision of this resume: one-page version <https://tinyurl.com/y8xdlfug>, complete version: <https://tinyurl.com/y2v59t36>
- **1 dan** on [CodeWars](#), ranked #111 on the whole site (Top 0.020%), primarily in Haskell, Agda, and Idris

---

## Publications & Preprints

- [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).