XueYan Zhang

+1 905-920-6053 xueyanzhang27@gmail.com Toronto
github.com/XueyanZhang linkedin.com/in/xueyanzhang27 xueyanzhang.github.io

- 3+ years AI NPU compiler with top-tier publications (ACL, EMNLP, COLM) toward AI full-stack expertise
- Unique compiler + NLP research background; Strong track record of fast 0-to-1 delivery

Experience

- **Heterogeneous Compiler Lab, Huawei Technology Canada** Toronto, Ontario
  Senior NPU Compiler Engineer *Sept 2022 - June 2026*
  - **NPU Compiler Optimization for LLMs**:
    - Spearheaded compiler optimizations in LLVM for featured networks, like **Flash Attention** and RMS norm, achieving **25%** improvement in operator performance and a 10% reduction in compiled binary size.
    - Served as the technical focal point for a small-sized team, leading communication and aligning technical progress with international counterparts.
    - Key features: Zero-overhead Hardware Loops, Ahead-of-Time Constant Propagation, Peephole Optimizations, Prefetching, and custom loop fusion.
  - **AI-powered Code Review Service**:
    - Built an AI-powered code review agent on **Jenkins**, integrating **Deepseek** as the LLM backend and **OpenCode** as the agent framework, with multi-agent collaboration for automated merge request analysis.
    - Configured GitLab CI pipelines and provided glab CLI tooling for MR view, diff, and API operations, enabling fully automated review feedback on every merge request.
    - Deployed Deepseek V4 Flash with 1M context window via **vllm** on two-node NVIDIA GB10.
  - **Unified Test Case Reduction Framework**:
    - Architected an award-winning test case reduction framework by unifying 5 open-source algorithms and pioneering a novel, LLM-based approach to auto-generate test properties.
    - This service is now used by **200+** developers and has boosted debugging efficiency by 50%, automatically minimizing a 10k-line test case to its 100-line root cause in under 30 minutes.
  - **Data Movement Interface Re-architecture**: Redesigned data movement interface, introducing LLVM intrinsics and MIR pseudos that eliminated error-prone code and simplified analysis for downstream passes.
  - **General Matrix Multiplication in MLIR**: Engineered a vectorization feature in MLIR to pattern match, transform, and optimize user codes to fully utilize the matrix computing unit. Supported double buffering.
  - **Student Intern Mentorship**: Mentored multiple software engineering interns on compiler projects, leading to the successful integration of their features into the production codebase and a final results presentation to headquarters.
- **UWaterloo CS136**, *Instructional Apprentice* Waterloo, Ontario
  Clarified complex concepts in algorithms and data structures for students in weekly office hours. *Sept 2021 - May 2022*
- **UWaterloo SWAG Lab**, *Undergraduate Research Assistant* Waterloo, Ontario
  Engineered research benchmarking automation with tracking, quantifying, and visualizing *Sept 2020 - Dec 2020*

Projects

- **OpenTikZ.org**, *Open-source TikZ library for AI-assisted academic diagrams* *May 2026*
  - Designed and shipped a community TikZ resource library, the "Flaticon for academic TikZ", delivering **100+** reusable assets for system, neural-network, and pipeline figures.
  - Shipped a **Claude Code** skill, enabling an AI agent to discover, edit, and recompile figures from natural-language requests so researchers produce publication-ready diagrams without hand-writing TikZ.
- **iMarkAi.com**, *Agentic bookmark with semantic search & web interactions* *Nov 2025*
  - Design, implement, and successfully published a commercial-grade bookmark extension on Chrome Web Store.
  - Architected intelligent search platform using **Pinecone** vector database and Google Gemini AI, achieving sub-second semantic retrieval across unstructured web data with 95%+ precision.
  - Implemented end-to-end **Stripe** billing for subscriptions and usage-based payments.
- **Tarot: Starry Guidance**, *End-to-end mobile app development* *Sept 2025*
  - Led the end-to-end, full-stack development on the WeChat mini-program platform; launched within 1 month.
  - Utilized web scraping for data; implemented HTML/CSS for UI, a cloud database, and an integrated LLM for personalized interpretations of drawn cards.
  - Conducted market research and competitor analysis; **reached 500+ users in first 3 months**.

Skills & Service

- **Programming**: C/C++, Python, Bash, Java/Kotlin, Unix/Linux, **git** Git
- **Toolbox**: Huggingface, Pytorch, LLVM, MLIR, Bazel, Docker, Claude Code/API/SDK, NVIDIA Spark

Education

- **University of Waterloo**, *Master of Math in Computer Science* — Waterloo, Ontario
  Research Stream in Software Engineering; Avg. 93/100; Graduate Excellence Award — *Jan. 2021 – Dec. 2022*
- **McMaster University**, *Bachelor of Engineering in Mechatronics Co-op* — Hamilton, Ontario
  GPA: 11.6/12; Dean's Honor List; Senate Scholarship — *Sept. 2015 – Apr. 2020*

Publications

- **Fusion R1: Empower PyTorch Graph Fusion via LLM with Reinforcement Learning** *Mar 2026*
  - Proposed an SFT+GRPO training framework for operator graph fusion in PyTorch Inductor, achieving avg. 15% end-to-end speedup (up to 3×) over torch.compile, outperforming Claude, GPT, and Gemini out-of-box. (Pending submission to NeurIPS)
- **IntentEval: Whether Large Language Models Answer What Users Actually Mean** *Jan 2026*
  - Current chat MT-Bench and AlpacaEval largely assume that users pose clear, complete, and single-intent questions. Real conversations are often messier: users omit context, use vague language, rely on false presuppositions. (Pending submission to EMNLP)
- **UORA: Uniform Orthogonal Reinitialization Adaptation in PEFT** *May 2025*
  - Investigated Parameter-Efficient Fine-Tuning (**PEFT**) methods, with a focus on **LoRA** and its variants.
  - Reproduced over 8 state-of-the-art artifacts; benchmarked performance across diverse tasks in Natural Language Processing (NLU, NLG, Instruction-tuning) and Computer Vision (ViT).
  - Proposed a novel method of interpolation-based reinitialization mechanism to achieve SOTA parameter efficiency, reducing trainable parameters by up to **15x** compared to LoRA while maintaining competitive performance.
  - *Accepted to Annual Meeting of the Association for Computational Linguistics* (**ACL 2025**)
- **Tiny Budgets, Big Gains: Parameter Placement Strategy in PEFT** *Aug 2025*
  - Introduced FoRA-UA, a super-efficient method that uses a sparse Fourier low-rank approximation with a universal shared adaptor, achieving state-of-the-art performance with only 1-5% of LoRA's parameters.
  - *Accepted to Conference on Empirical Methods in Natural Language Processing* (**EMNLP 2025**)
- **Large Language Model is not a (Multilingual) Compositional Relation Reasoner** *Jan 2024*
  - Pioneered a comprehensive, multilingual evaluation of LLM compositional reasoning by creating and open-sourcing the **MCR Benchmark**. This new benchmark systematically assesses model performance across six distinct relation categories in five languages (English, Chinese, French, Japanese, and Korean), identifying and quantifying critical reasoning deficiencies in state-of-the-art models.
  - *Accepted to Annual Conference on Language Modeling* (**COLM 2024**)
- **Word Semantics Consistency in Language Models** *May 2024*
  - Quantified the internal-external knowledge mismatch for word semantics across Encoder, Decoder, and Encoder-Decoder LMs.
  - **Probing internal states** versus querying model outputs on 3 benchmarks (word similarity, Named Entity Recognition (NER), analogy) revealed that probes access significantly more accurate semantic knowledge, highlighting a key representational discrepancy.
- **On the Caching Schemes to Speed Up Program Reduction** *Jan 2022*
  - Master's Research. Program reduction minimizes the failing test case while preserving the bug-triggering properties.
  - Proposed Refreshable Compact Caching (**RCC**) by leveraging the characteristics of monotonically decreasing subset in program reduction, which is a both computational and memory-efficient caching scheme for program reduction.
  - Effectively reducing **64%** redundant property-test queries, accelerating up to **43%** in reduction time, and minimizing memory footprint by **99.97%** from 100+ GB (text-based) to 6 MB (RCC).
  - *Accepted to ACM Transactions on Software Engineering and Methodology* (**TOSEM 2024**)

Peer Review & Community Service & More

- Reviewer, International Conference on Learning Representations (ICLR 2025); ACL ARR 2026
- Delivered Prompt Engineering, Reinforcement Learning, and Fine-tuning Crash Courses
- PC hardware enthusiast for ultra-compact ITX builds