

Yuma Oda

Research Engineer | ML Systems, Compilers, Inference

✉ yuma.oda@keio.jp

🐙 github.com/hikettei

Research Engineer at Mirai Labs and second-year undergraduate at Keio University. My work spans ML systems, compilers, and efficient LLM inference, with a focus on speculative decoding for Apple devices. Numerical computing enjoyer and Common Lisp enthusiast.

RESEARCH INTERESTS

Fascinated by the computational capabilities of AI and by the design of high performance training and inference systems, from models and compilers to networking.

EDUCATION

Keio University | *B.A. Candidate, Environment and Information Studies*

Apr 2025 - Apr 2029 (Expected)

Focus: Computer Science, Transactions, and Databases

EXPERIENCE

Mirai Labs | *Research Engineer*

Mar 2026 - Present

Full-time · Remote · San Francisco, CA, USA

- Co-developed **DFlash-TfM**, a state-of-the-art speculative decoding method, achieving a $4.37\times$ speedup over autoregressive decoding and 24.7% higher throughput than optimized DFlash on Qwen3.6-27B (BF16, single NVIDIA B200).
- Co-developed **speculative decoding in Uzu** for Apple M5-series Macs, achieving nearly $2\times$ the decoding throughput of MTPLX and over $3\times$ that of llama.cpp on Qwen3.6-27B at comparable quantization levels.

Aoha, Inc. | *Software Engineer*

Dec 2023 - Aug 2026

Part-time · Hybrid · Minato-ku, Tokyo, Japan

- Built deep-learning compiler infrastructure and neural network quantization methods for computer vision models.
- Researched efficient training methods for mixture-of-experts models.
- Built reinforcement learning infrastructure across 56 NVIDIA H200 GPUs for automated RTL design, including agent architecture, training harnesses, and data/evaluation pipelines.

PUBLICATION

Yuma Oda, Ryan Mathieu, Roman Knyazhitskiy, and Artur Chakhvadze. “Trees from Marginals: Autoregressive drafting with factorized priors.” *arXiv preprint arXiv:2607.06763*, 2026.

PERSONAL PROJECT & OPEN SOURCE CONTRIBUTION

Caten | *Creator and Maintainer*

- Created an experimental deep-learning compiler in Common Lisp using lightweight IRs, polyhedral compilation, and an optimizing pattern matcher.
- Designed a RISC-style graph IR that lowers computation to 26 composable primitive operations.
- Built lazy evaluation, Clang JIT compilation, model inference demos, and experimental training and automatic differentiation support.

tinygrad | *Fast-math approximations, PR 5187*

- Completed an approximation bounty by implementing SIN, LOG2, and EXP2 approximations; merged into tinygrad.

HONORS & AWARDS

METI Minister’s Award, U-22 Programming Contest · \$3,600 award
Sokan no Susume Scholarship, Keio University · \$15,000 scholarship

Nov 2023

Apr 2025 - Apr 2029 (Expected)

LANGUAGES

Japanese: Native **English:** Upper-intermediate (CEFR B2)