

MOSTAFA HANY

mstfa.hanny@gmail.com | +201028322108 | Giza, Egypt
catb1t.github.io | linkedin.com/in/catbit/ | github.com/CatB1t

EXPERIENCE

Software Engineer

Unify

Feb 2025 - Apr 2026

| Remote

- Owned the development and maintenance of the Python SDK for the Unify API.
- Increased logging throughput by $10\times$ by redesigning the pipeline using asynchronous (asyncio-based) processing, reducing overhead in high-throughput environments.
- Reduced system initialization time by $4\times$ through improved dependency loading strategies.
- Built and extended caching systems, including semantic caching for LLM interactions, reducing redundant requests, latency, and compute cost.
- Implemented context compression and time-awareness mechanisms for long-running LLM workflows, enabling efficient handling of extended conversations within token constraints.
- Refactored core agent execution loop into a modular, maintainable architecture, enabling easier extensibility and faster iteration on new features.
- Diagnosed and resolved nondeterministic prompt behavior that led to large-scale cache inconsistencies, improving correctness and cache efficiency.
- Expanded multi-provider LLM support by migrating to a LiteLLM based abstraction layer.
- Developed a cost tracking infrastructure for LLM usage, improving visibility and control over API spend.
- Established a testing infrastructure (custom fixtures and decorators) to improve the reliability, debuggability, and execution speed of test suite.
- Designed and implemented end-to-end tracing and metrics for LLM requests, improving observability, debugging speed, and system reliability.

Software Engineer (Computational Geometry)

Atomica

Sep 2024 - Feb 2025

| Remote

- Created comprehensive visual examples to demonstrate library usage, enabling developers to easily understand and effectively utilize the API.
- Enhanced CI workflows to ensure more stable releases and a streamlined development process.
- Integrated run-time sanitizers to enhance code reliability and debug various issues.
- Incorporated static analysis tools into the code base, significantly improving code quality and reliability.
- Optimized core functions, improving execution time from 5s to 1s, leading to a $5\times$ speedup.

Software Engineer

Unify

Aug 2022 - Jan 2024

| Remote

- Managed a team of five engineers.
- Designed and implemented a proof-of-concept for a source-to-source transpiler to convert code between different deep learning frameworks, e.g., TensorFlow to PyTorch.
- Redesigning the testing framework to improve modularity and readability, achieving a 50% reduction in code size and making code multiprocessing friendly.
- Designed and implemented functionality to support the import of multiple instances of the framework with different backends at runtime, reducing the average test runtime from 120 to 10 seconds.
- Designed and implemented an API backend generation feature using Python AST (Abstract Syntax Trees), allowing developers to generate a new API backend with only a few lines of configuration.
- Extended the integration of property-based testing in the testing framework to specifically meet our requirements with less runtime overhead and reduced code.
- Extended and maintained the Ivy framework support on Windows.
- Added support for GPU testing in the testing framework.

PUBLICATIONS

Accelerating Deep Learning using Ivy | Link NeurIPS, 2023
Guillermo Sanchez-Brizuela, Ved Patwardhan, Matthew Barrett, Paul Anderson, **Mustafa Hani**, Daniel Lenton.

RELATED SKILLS

Programming Languages: C++, C, GLSL, Python, Lua, Bash
Libraries / Frameworks: OpenGL, Nanobench, doctest, ImGui, SFML, Assimp, GLM, GLFW, SDL
Dev Tools: RenderDoc, CMake, Tracy, Premake, Docker, git, gdb

EDUCATION

Faculty of Computers & Artificial Intelligence - Cairo University Giza, Egypt
Bachelor of Science degree in Computer Science - Artificial Intelligence Major Sep 2024

PERSONAL PROJECTS

Software Raytracer | Link C++, Lua C API, STB, Premake
Cross-platform software raytracer from scratch in C++. Supports lighting, shadows, and reflections.
Embedded Lua C API for scene configuration using a Lua file.

Aker (3D Renderer) | Link C++, OpenGL, ImGui, Assimp, SDL, CMake
Cross-platform 3D renderer with OpenGL. Supports lighting, shadows, and model loading. Full scene editor with ImGui.

CS:GO Object Detection | Link Python, YoloV5, PyTorch, OpenCV
Detecting CS:GO objects using YOLOv5 with a custom OpenCV annotation script.

The Office (Game) | Link C#, Unity
Brackey's Game Jam 2021 submission. Built with Unity in 6 days.