Xianglong Hu

San Francisco, CA

Email : hxianglong@gmail.com
Mobile : +1-206-671-8930

Publications

Distilling Tool Knowledge into Language Models via Back-Translated Traces 115 stars on github: Accepted at the **ICML 2025** Workshop on Multi-Agent Systems (MAS). Developed an innovative methodology to translate tool-based reasoning into natural language reasoning via back-translated traces.

TCP: A Benchmark for Temporal Constraint-Based Planning huggingface: Accepted at **ACL/EMNLP 2025**. Introduced a specialized benchmark and synthetic dataset to evaluate and train LLMs in complex, time-sensitive planning and scheduling tasks.

Loong: Verifiable Long-form Reasoning Framework 485 stars on github: Developed as a scalable synthetic data generation and validation pipeline to strengthen LLM reliability in logic-intensive domains. Trained Qwen with generated data on verl, which reproduced deepseek results.

Experience

Amazon Web Services — San Francisco, CA

Software Development Engineer II — *Nov 2024 - Present*

- **Scalable Workflow Orchestration**: Led the architectural refactoring of the Outpost infrastructure provisioning service, transitioning to a distributed workflow model using AWS Step Functions and Lambda. Improved system scalability and significantly reduced maintenance overhead through decoupled, stateful execution.
- **Lifecycle Automation**: Spearheaded the end-to-end design and development of the Outpost Decommission Service, automating complex hardware retirement protocols and ensuring secure, compliant infrastructure turnover.
- **Edge Reliability**: Maintained and optimized the Snow OTA (Over-the-Air) Update Service, ensuring high-availability and seamless firmware deployments across globally distributed edge computing devices.

Camel-AI — Remote

Open Source Developer and Researcher — *Nov 2024 - Present*

- **Scaled LLM Reasoning (Loong)**: Engineered an open-source framework for synthesizing and verifying long-form Chain-of-Thought (CoT) data at scale. Applied Reinforcement Learning on logic data to reproduce state-of-the-art reasoning results of DeepSeek-R1 and Logic-RL, inducing emergent behaviors like self-reflection and verification. 15.8k stars on github
- **Reinforcement Learning (ReaL-TG)**: Involved in the development of the ReaL-TG framework, which utilizes RL to optimize language models for explainable link forecasting on temporal graphs. Contributed to designing reward signals that prioritize transparency and logical consistency in model predictions.
- **Supervised Fine-Tuning (Magenta)**: Leveraged SFT workflows to distill tool-use knowledge into compact models via back-translated traces, enabling high-performance autonomous agent capabilities with significantly reduced inference overhead.
- **Temporal Planning & Benchmarking (TCP)**: Designed and published the TCP Benchmark, a specialized evaluation suite for measuring LLM performance on temporal constraint-based planning, bridging a critical gap in multi-step reasoning assessment.
- **Multi-Agent Orchestration**: Contributed to the CAMEL-AI open-source ecosystem, focusing on autonomous communication protocols and the deployment of “societies” of LLM agents to solve complex, distributed tasks.

Open Source — Remote

Open Source Developer — *Oct 2025 - Present*

- **Nano-vLLM Architecture Profiler**: Conducted a deep-dive architectural analysis of the Nano-vLLM (1.2k LOC) inference engine, mapping the lifecycle of PagedAttention and KV-cache block management.
- **Simulated Performance Benchmarking**: Engineered a Mock Execution Engine to simulate GPU kernel latency, enabling the testing of scheduling algorithms and memory fragmentation on CPU-only environments.
- **Optimization Research**: Identified potential throughput bottlenecks in the FCFS scheduler during long-form reasoning (CoT) generation; documented architectural improvements for multi-agent orchestration.
- **KV-Cache Visualizer**: Developed a Python-based visualizer for tracking KV-Cache block allocation and reference counting in real-time.

CloseFactor — Jersey City, NJ

AI Software Development Engineer — *Mar 2023 - Jun 2024*

- **LLM Product E2E Delivery**: Led and developed Account Plans from concept to successful implementation, utilizing cutting-edge LLM techniques such as OpenAI, rerank, RAG, and model fine-tuning with LoRA. Architected a robust system incorporating unreliable components like scraper services.
- **Agile & Lean Startup**: Conducted customer interviews and rapidly iterated products based on feedback. Collaborated closely with Product Management to ensure technical and product alignment.
- **Impact**: Spearheaded the development of the most critical product, contributing to nearly 100% of new ARR. Sold over 300k ARR. Developed core recommendation products (News & Contact Recommendation, SmartSend) with LLM. Drove 130% over engagement and 350% in logins. 30 positive reviews on G2.

Ads & Promotion, DoorDash — Seattle, WA

Software Development Engineer — *Aug 2022 - Nov 2022*

- **Cache Optimization & Cost Reduction**: Designed and implemented an in-memory cache for merchant metadata to reduce Redis traffic. Refactored the data retrieval path. Reduced AWS Redis cost by 90% and AWS ECR Cost by 50%. Reduced latency of all tier-1 APIs by 50%. Work is recognized by staff engineer on LinkedIn.
- **Unit Test Optimization**: Identified code build issues and optimized unit test setups. Slashed code build time by 60% (from 15 min to 4 min). Improved team productivity.

IoT Core, Amazon Web Services — Seattle, WA

Software Development Engineer — *Apr 2020 - Aug 2022*

- **Full Cycle Product Launch**: Designed and led the development of Fleet Metrics, a product to monitor IoT devices' state over time in CloudWatch. Built and tested a new customer-facing API which emits metrics periodically. Technical stack included AWS EC2, Elastic Search, Dynamo, CloudWatch and SQS.
- **Event-Driven Pipeline**: Designed and implemented a distributed stateless event processing microservice. Participated in the full cycle of the product feature launch from scoping, designing, implementing, testing and releasing.
- **Oncall Automation**: Automated all oncall SOPs with python scripts. Reduced oncall time from 5 minutes to under 5 seconds. Reduced the latency of one customer-facing API by 90%. Implemented a mechanism controlling system throughput with dynamic config.
- **Documentation & Mentorship**: Contributed to team documentation. Maintained onboarding wikis for new hires and oncall processes.

Onai Technology — New York City, NY

Software Engineer Internship — *Jul 2019 - Dec 2019*

- **Blockchain & Accumulator**: Benchmarked a batching accumulator. Implemented and designed a Merkle-tree like membership-proof with Celo pairing groups in Rust.
- **Git as NoSQL (Rust)**: Designed a key-value store with git to achieve a version-controlled NoSQL database with minimal dependencies. Utilized git internal data structures like blob, tree, and reference to store objects while using path as key. Implemented and tested in Rust with git2.
- **Domain-Specific Language (Idris)**: Designed an LL(0) domain-specific language for state machines, compiled into Idris to reduce repetitious skeleton code and JavaScript to visualize a state machine. Grammar verified in Antlr. Implemented a VS Code frontend for Idris.

Experiment Center for Physics, Fudan University — Shanghai, China

Research Assistant — *Jul 2018 - Nov 2018*

- **Acoustic Wave Apparatus**: Designed and implemented an apparatus for acoustic wave generation and experiment. Measured and analyzed acoustic properties on Android smartphones for pedagogical purposes, primarily Fourier transformation of acoustic waves, real-time data processing and rendering.
- **Android Development**: Designed the frontend with interactive charts and experiment parameter settings, and backend in Android. Interacted with hardware by thread programming. Data export supported.
- **Publication**: Research results published in Physics Experiments, funded by the National Natural Science Foundation (China).

Laboratory of Computer Vision and Machine Learning, Fudan University — Shanghai, China

Research Assistant — *May 2017 - Dec 2017*

- **Hierarchical Bayesian Neural Network**: Mathematically formulated a hierarchical Bayesian neural network with variational inference and adapted it to incremental learning to eliminate catastrophic forgetting when new classes appeared. Network of Gaussian probabilistic weights implemented with local reparametrization and practical Monte Carlo in PyTorch. Trained with Cuda GPUs. Github.

Atomic and Optical Physics Lab, Fudan University Shanghai, China
Research Assistant *Jul 2015 - Jul 2017*

- **Quantum Wave Reconstruction**: Adapted various optimizations including genetic algorithm and Powell's method to the reconstruction of quantum waves from nonlinear noises in MATLAB.

Projects

Nano-vLLM Architecture Profiler: Conducted deep-dive architectural analysis of the Nano-vLLM inference engine. Engineered a Mock Execution Engine to simulate GPU kernel latency. Developed a Python-based visualizer for tracking KV-Cache block allocation and reference counting in real-time.

ReaL-TG: Reinforcement Learning for Temporal Graph Forecasting: Designed an RL framework enabling LLMs to perform explainable link forecasting on temporal graphs. Utilized GRPO and outcome-based reward system. Proposed Penalized Mean Reciprocal Rank (pMRR) and LLM-as-a-Judge evaluation. Developed T-CGS algorithm for temporal subgraph selection. ReaL-TG-4B outperformed larger frontier models on both seen and unseen graphs.

Distilling Tool Knowledge into LLMs (Magenta): Developed a SOLVER AGENT interleaving natural language planning with SymPy toolkit calls. Engineered a Back-Translation Pipeline using specialized LLM agents (Translator, Judge, Rephrase). Fine-tuned Qwen2.5-Math-7B on 11.6k synthesized traces. Achieved significant gains on competition-level benchmarks (AIME, AMC) with tool-free deployment.

Loong: Scaling Synthetic Reasoning Data: Built LOONGBENCH, a human-vetted seed dataset of 8.7k examples across 12 reasoning-intensive domains. Designed LOONGENV, a modular environment using multi-agent workflows and code execution to synthesize verifiable question-answer pairs. Reproduced DeepSeek-R1 and Logic-RL results with RLVR on logic-heavy datasets.

TCP: Temporal Constraint-Based Planning Benchmark: Developed a scalable pipeline to generate 600 naturalistic dialogue-based problems involving asynchronous task scheduling, time zones, and dynamic availability. Implemented a symbolic verification system (Hard Check). Benchmarked 20+ state-of-the-art models revealing gaps in asynchronous planning and time-zone reasoning.

Wolfram Language Server 200 stars on Github: A language server protocol implementation for Wolfram Mathematica. Provided grammar diagnostics, hover, completion, and resolving features. Implemented in Wolfram Mathematica in a functional programming paradigm with immutable data. Client in JavaScript/TypeScript released on VS Marketplace. Communication via socket through JSON-RPC protocol. Compatible with any LSP-supporting client like Emacs and Vim.

Education

New York University New York City, NY
Master of Science in Computer Science *Jan. 2018 – Jan. 2020*

Fudan University Shanghai, China
Bachelor of Science in Physics *Sep. 2013 – May. 2017*

Programming Skills

Languages: Kotlin, Java, Python, Rust, Wolfram Mathematica, MATLAB, JavaScript, TypeScript
Technologies: LLM, PyTorch, RAG, Langchain, Prompt Engineering, AWS, GCP, Sql, NoSql, Restful API, Distributed Systems, CI/CD, DevOps, Linux, Spring, Guice, Micronaut, AI Agents, Vibe Coding, verl, Reinforcement Learning, PagedAttention, vLLM, Android