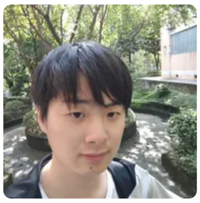
Jiale Zhao

Male | Native place： Beijing | 18256731893 | jialeuuz@gmail.com



Education

Chongqing University of Posts and Telecommunications Bachelor Algorithm engineer 2021-2025

Initiated ML research in a university lab (Fr), conducted CV research (So. Fall), transitioned to NLP research (Jr. Spring), and gained industry experience as an LLM Algorithm Intern at Li Auto (Jr. Year).

Professional Summary

I have two years of LLM algorithm internship experience, with strong coding and research skills in LLM and multimod al domains. **My Key interests are following:**

1. Agent-based LLMs **2. Data and Self-improving** **3. Human-LLM Interaction**
4. Interpretability and Analysis **5. Bench and Evaluation** **6. Bridging Research and Application**

Publications Under Review

ThinkPilot 2025.05-now

Submitted to AACL 2025 via ARR (plan to resubmit to ACL 2025 after revision based on reviews)

Personal Contributions: 1. Designed experiments and implemented framework 2. Introduced explainability methods into iterative algorithm 3. Wrote appendix and contributed to main text

Research experience

LLM Algorithm Intern LI Auto 2023.09-now

Data Flywheel for Code LLM

This involves a cycle of SFT, evaluation, production training data, data filtering, and then back to SFT.
I have proposed an iterative framework based on the aforementioned principles. This framework centers around evaluation, combining data production and data filtering to mass-produce high-quality training data and high-quality evaluation data.

Multi-step Reasoning and Tool Invocation Agent Based on Code LLM

Construct SFT data for LLM Q&A tasks, implement API function calls, and solve complex reasoning problems through multi-step processes.

MindGPTo

An end-to-end multimodal application that replicates much of GPT-4O's functionality while incorporating unique paralinguistic features.
a. Built the MindGPTo from scratch with a modular design and front-end/back-end separation.
b. Established extensive data production pipelines for bulk audio data generation. Conducted sft on models used by mindGPTo, focusing on enhancing conversational and anthropomorphic capabilities.

Potential Research Topics

Agent-based LLMs

Efficient Complex Problem Solving

Many real-world problems require reasoning across multiple sources and dynamically incorporating user feedback. For example, planning a weekend trip to Beijing requires gathering date and location information, retrieving travel guides and candidate itineraries, then refining options based on weather and transportation. Throughout the process, users may add constraints such as a preference for nature sites, requiring the agent to update plans on the fly instead of starting over.

To address these challenges, my ideas include:

1. Decomposing complex tasks into simpler sub-tasks that can be handled more efficiently.
2. Introducing a **time-frame** mechanism to dynamically accept user updates and regularly reflect on the agent's progress and decisions.
3. For basic sub-tasks, leveraging multiple parallel smaller models (e.g., several 7B LLMs) with time-frame coordination can achieve higher efficiency and accuracy compared to a single large model under the same computational cost.

Data and Self-Improving

Self-Evaluation → Self-Improvement

During my early internship work on LLMs for code generation, I observed that while LLMs may initially fail to solve certain problems, they can often arrive at correct solutions when provided with various forms of guidance. This led me to consider whether LLMs could leverage self-evaluation and reflection mechanisms to iteratively improve themselves.

For code tasks, I have identified three typical failure modes: (1) problems that LLMs solve inconsistently; (2) problems rarely solved spontaneously but solvable through simple hints; and (3) problems requiring complex or even unattainable guidance. The first type can be addressed through increased sampling; the second via heuristics or self-guided strategies; and for the third, iterative evolution may gradually improve performance (without introducing external models). However, continuous self-improvement requires large and diverse data, and standard benchmarks are insufficient. As models get better, fresh, challenging data is needed and rigorous filtering is required to ensure data quality. Designing diverse and accurate test queries is often easier for LLMs than solving the tasks themselves, but ensuring the diversity and correctness of such tests remains a core challenge. Given my focus has shifted from code recently, I recognize the need for further research in this area to identify state-of-the-art solutions.

Human-LLM Interaction

Enabling LLMs to Ask Questions

Current LLM-user interactions face several challenges: (1) LLMs often exhibit excessive agreeableness, affirming user statements even when incorrect; (2) due to this, when facing underspecified or ambiguous queries, LLMs guess user intent rather than seeking clarification, leading to unsatisfactory responses—**despite user underspecification being common in practice but rare in standard evaluation benchmarks;** (3) for difficult tasks, LLMs rarely solicit missing information from users, even though effective user input can greatly reduce task difficulty.

To address these issues, my ideas are: (1) enabling LLMs to recognize and challenge incorrect or contradictory information in user queries; (2) equipping LLMs to detect and explicitly **request missing conditions** for underspecified problems; (3) guiding LLMs to summarize key challenges for hard tasks and proactively **seek user assistance when needed.**

Interpretability and Analysis

Insight → Control → Safety

When exploring thought-chain interpretability of LLMs, I became interested in safety issues. I observed that most current attack and defense methods operate at a shallow level, focusing on surface cues when prompting errors or defenses. In my view, **only by leveraging deep interpretability to understand the inner mechanisms of LLMs can we develop more effective means to control the model's reasoning process**—leading to both safer control and more efficient testing of vulnerabilities.

Building on thought-chain analysis, I have noticed some recent work demonstrates that training small auxiliary models, without modifying LLM weights, can **"overclock" the model by shortening its reasoning chains.** Inspired by this, I am considering whether, by **constructing comprehensive query sets and analyzing large numbers of model responses, we can train control modules**—much like a psychologist shaping thoughts—to achieve targeted guidance over the model's reasoning (for example, by identifying and applying the right direction vectors to influence its process).

Bench and Evaluation

How Can We Make Benchmarks "Dumber"?

Most current benchmarks are too "smart"—**they present queries that are concise, fully specified, and unambiguous, which rarely reflects real-world usage.** For example, mathematical or coding benchmarks assume well-formed, complete problem statements. However, in realistic scenarios, even experts often struggle to formulate perfect queries outside their domain; user input is typically incomplete, ambiguous, or verbose, and resolving intent often requires multi-turn dialogue with an LLM to clarify needs and achieve satisfactory responses.

To address this, I propose designing a framework to transform existing benchmarks into "dumber" versions that better simulate real user interactions. Specifically, this involves turning a single benchmark query into a simulated "user" engaging in a multi-turn conversation with the LLM. For logic-heavy tasks like math and code, this includes introducing errors, omitting conditions, and making the queries more colloquial and verbose. For knowledge-based queries, such as in medicine, queries can be intentionally vague or incomplete (e.g., simulating a user withholding embarrassing symptoms), thereby requiring the LLM to elicit and clarify missing information. Such benchmarks would enable more realistic evaluation of LLMs' ability to handle underspecified and ambiguous user input.

Bridging Research & APP

Academia Must Not Be Disconnected from Real-World Impact

Having spent several years immersed in LLM research, I have had the privilege of working with people from both academia and industry—at my university and during my internships. These experiences allowed me to appreciate the distinct mindsets and approaches in each sector, and further solidified my commitment to an academic path.

Since before college, I have been passionate about AI, motivated by the belief that one cannot always rely on others to change the world. Driven to make an impact myself, I sought out opportunities to study AI, joining a deep learning lab early in my undergraduate years. In my pursuit of the right research direction, I worked in three different labs and, after the release of GPT-3.5, decided to focus on NLP. Because of limited academic resources, I also pursued long-term industry internships—spending nearly half of my college years balancing academia and industry.

Through these diverse experiences, I have mapped out a future path: pursue a PhD for deeper and freer research, seek a faculty position to explore my academic interests, and cultivate collaborations or entrepreneurial efforts with industry to overcome practical limitations like computational resources. All of these steps, however, ultimately serve one goal: ensuring that AI research is meaningfully connected to real-world needs and truly benefits society.

Addendum – Publications & Dataset Update

Under Review:

Decoding the Ear: A Framework for Objectifying Expressiveness from Human Preference Through Efficient Alig

arXiv: https://arxiv.org/abs/2510.20513

Dataset:

ExpressiveSpeech — High-quality bilingual (Chinese–English) expressive speech dataset (~51h, ~14k utteranc

Hugging Face: https://huggingface.co/datasets/FreedomIntelligence/ExpressiveSpeech

Project page: https://freedomintelligence.github.io/ExpressiveSpeech/