

# Yuangang Li

[yuangang@usc.edu](mailto:yuangang@usc.edu) | (+1) 213-421-9602 | [Linkedin](#) | [Github](#) | [yuangang.tech](#) | [Google Scholar](#) | Los Angeles, CA

## RESEARCH INTERESTS

Trustworthy AI(LLM), ML Systems(LLMSys), AutoML, LLM, Federated Learning, AIoT, Edge AI, Cloud Computing

## EDUCATION

**University of Southern California** | M.S. in Computer Science (GPA: 3.82/4.0) 01.2022 - 05.2024

**Beijing City University** | B.S. in Software Engineering (GPA: 3.6/4.0, Top: 1/150) 09.2017 - 07.2021

## PUBLICATIONS (\* co-first author)

- [1] **NLP-ADBench: NLP Anomaly Detection Benchmark** (Submitted to WWW 2025). Authors: **Yuangang Li\***, Jiaqi Li\*, Zhuo Xiao\*, et.al., [Yue Zhao](#)
- [2] **CausalCore: Reducing Hallucinations in LLMs through Enhanced Causal Inference Capabilities** (In Progress; To be submitted to ICML 2025). Authors: **Yuangang Li**, [Yue Zhao](#)
- [3] **AD-LLM: When Anomaly Detection Meet Large Language Models** (Submitted to ACL 2025). Authors: Tiankai Yang, Yi Nian, **Yuangang Li**, et.al., [Yue Zhao](#)
- [4] **Towards More Accurate US Presidential Election via Multi-step Reasoning with Large Language Models**([arxiv](#)). Authors: Chenxiao Yu, Zhaotian Weng, **Yuangang Li**, Zheng Li, [Xiyang Hu](#), [Yue Zhao](#)
- [5] **H-FedSN: Personalized Sparse Networks for Efficient and Accurate Hierarchical Federated Learning for IoT Application** (Submitted to INFOCOMM 2025). Authors: **Yuangang Li\***, [Jiechao Gao\\*](#), [Yue Zhao](#), [Brad Campbell](#)
- [6] **FedMetaMed: Federated Meta-Learning for Personalized Medication in Distributed Healthcare Systems** (BIBM-AIBH 2024). Authors: [Jiechao Gao](#), **Yuangang Li**
- [7] **FedLDR: Federated Local Data-infused Graph Creation with Node-centric Model Refinement** (ICDM-SSTDM 2024). Authors: [Jiechao Gao](#), **Yuangang Li**, Syeda Faiza Ahmed
- [8] **Artificial Intelligence-Aided Digital Twin Design: A Systematic Review** ([Preprints 2024](#)). Authors: Nan Hao\*, **Yuangang Li\***, Kecheng Liu\*, et.al., [Tianfan Fu](#), [Yue Zhao](#)
- [9] **FedBCGD: Communication-Efficient Accelerated Block Coordinate Gradient Descent for Federated Learning** (ACM MM 2024). Authors: Junkang Liu, Fanhua Shang, Yuanyuan Liu, Hongying Liu, **Yuangang Li**, YunXiang Gong

## RESEARCH EXPERIENCES

**University of Southern California** | Research Assistant (Advisor: Prof. [Yue Zhao](#)) 07.2023-Present

**NLP Anomaly Detection Toolkit and Benchmarking Development** | LLM, AutoML, NLP, Anomaly Detection, PyTorch

- Created a robust benchmarking system for evaluating algorithmic efficacy with over 40 PyOD library O.D. algorithms and 3 end2end algorithms across 20 datasets to facilitate comprehensive performance analysis and comparison<sup>[1]</sup>.
- Redesigned NLP datasets into a unified format for anomaly detection, addressing a field gap, and developed a system to automatically recommend optimal AD algorithms based on benchmarks<sup>[1]</sup>.
- Co-first authored a systematic review on AI-aided digital twin design, analyzing how machine learning enhances digital twins and their applications across multiple domains<sup>[8]</sup>.
- Constructed causal reasoning datasets, fine-tuned LLMs, and evaluated using hallucination benchmarks<sup>[2]</sup>.
- Investigated generative AI techniques to tackle AD problems<sup>[3]</sup> and explored the capabilities of LLMs in politics<sup>[4]</sup>.

**University of Virginia** | Research Collaboration (Advisor: Prof. [Yue Cheng](#)) 09.2024-12.2024

**Effective Machine Learning Systems Development** | LLM Inference, LLM compression

- Developed an automated method to generate optimally compressed LLMs, focusing on enhancing model accuracy and deployment efficiency through innovative quantization and calibration techniques.

**University of Virginia** | Independent Researcher (Advisor: Dr. [Jiechao Gao](#)) 06.2024-10.2024

**Federated Learning System Development** | Federated Learning, PyTorch, Python, Spatio-temporal data

- Developed the FedMetaMed, integrating federated learning and meta-learning to enhance personalized medication strategies across distributed healthcare systems, improving model adaptability and privacy preservation<sup>[6]</sup>.
- Single-handedly developed FedLDR, a brand new federated learning algorithm that employs GCN to enhance spatio-temporal data analysis through local data integration and node-centric optimization<sup>[7]</sup>.
- FedMetaMed<sup>[6]</sup> and FedLDR<sup>[7]</sup> were accepted by BIBM and ICDM respectively, and I gave talks on these research.

**University of Virginia** | Independent Researcher (Advisor: Dr. [Jiechao Gao](#), Prof. [Brad Campbell](#)) 12.2023-07.2024

**Hierarchical Federated Learning System Development** | Hierarchical Federated Learning, PyTorch, Python, Sparse Network

- Independently developed H-FedSN pushes the boundaries of IoT with a unique approach that uses masking techniques to train a sparse network, enhancing personalization through client-based transfer learning. Applied to non-IID IoT datasets, it achieves high accuracy and boosts communication efficiency by at least 58x<sup>[5]</sup>.
- Solely developed and integrated innovative federated learning algorithms—FedAvg, FedCAMS, FedPer, PerFedAvg, and FedRS—into a hierarchical framework to optimally benchmark against H-FedSN<sup>[5]</sup>.

**Xidian University** | Research Collaborator and Co-author 12.2023-04.2024

**Federated Learning System Development** | Federate Learning, PyTorch, Python, LLM

- Proposed FedBCGD, the first to use block communication in training large models, enhancing training speed with distributed tech. Implemented TOPK, FedAdam, and others, benchmarking them against FedBCGD's performance<sup>[9]</sup>.

**University of Southern California** | Research Developer (Advisor: Dr. [Iordanis F](#)) 03.2023-07.2023

## **Distributed ML Execution Framework Development** | *MLSys, AutoML, Ray, Docker, Github Action, PyTorch, Pytest*

- Contributed to "[Ablator](#)", an open-source Deep Learning framework used by 40+ USC researchers for horizontal scaling of ablation experiments and hyperparameter tuning, encompassing 70 pull requests.
- Implemented distributed experiment execution with **Ray**, managed open-source projects, set up **CI pipelines** via GitHub Actions, oversaw release management and version control, and authored **pytest** unit tests with **97%** coverage.
- Solely launched 'python-rclone' on PyPI, a Python API for RClone that streamlines cloud data synchronization for 'Ablator', removing pre-installation requirements and enabling automatic binary selection ([python-rclone](#)).

## **Chinese Academy of Sciences** | Research Assistant (Advisor: Prof. [Guoquan Wu](#))

05.2021-09.2021

### **Automated Testing Platform Development** | *Docker, Node.js, JSON, Vue.js, RobotFramework*

- Contributed to the R&D of a web-based automated testing tool using Record and Playback technology, significantly enhancing test case management by enabling streamlined recording, editing, execution, analysis, and result generation. This implementation boosted end-to-end testing efficiency by 300% and saved over 15 hours per week.
- Independently developed a script parser using Node.js that converts user actions recorded in JSON format into executable Robot Framework and Selenium scripts, enabling the replay and repeated execution of these user actions.
- Single-handedly created innovative UI components using Vue.js and AceEditor, orchestrated the optimal containerization of the program with Docker, and automated the DevOps pipeline to maximize development efficiency.

## **WORK EXPERIENCES**

### **SenseTime** (Top Tier AI Company) | Infrastructure Engineer & Researcher

01.2022-01.2023

#### **SaaS Platform Development** {[Demo](#)} | *Kubernetes, Docker, Go, CRD, Operator-SDK, Helm3, Prometheus, Grafana*

- Developed "RocketMQ as a Service", akin to "RabbitMQ as a Service" in AWS Marketplace, offering fully managed SaaS-based RocketMQ clusters, increasing **100%** creation speed and saving **10+ hours/week** in manual operations.
- Utilized Operator SDK to build a Kubernetes-based RocketMQ Operator and CRD automating lifecycle management.
- Employed **Helm3** to package RocketMQ's components into Helm charts, simplifying Kubernetes deployment.
- Implemented **Prometheus** and **Grafana** for real-time monitoring of critical service metrics and node health.
- Automated workflows, including unit tests, image builds, and Helm3 Chart updates, via GitLab CI/CD.
- Researched and evaluated container runtimes (sysbox, crun, youki) for suitability as replacements for Docker in the SaaS platform, ensuring CRI-O compliance and robust community support.
- Optimized a machine learning training pipeline using GPUs on Kubernetes for enhanced computational efficiency.

### **Xiaoni Translations (Beijing) Technology Co., Ltd.** | FullStack Developer

01.2021-04.2021

#### **Text Translation Platform Development** | *Java, SpringBoot, Spring, Java Persistence API, Maven, Nginx, MySQL, Git*

- Developed an AI document translation system with Java/Spring/Maven, independently created a PDF/XML parsing module attracting 30,000 MAUs, used Nginx for reverse proxy, and managed version control with Git.

## **OPEN-SOURCE CONTRIBUTIONS**

### **Apache/rocketmq-operator** {[Github Link](#)} | *Kubernetes, Docker, Go, CRD, Operator-SDK*

12.2023-01.2024

- Resolved process memory allocation inaccuracies in Kubernetes pods by replacing the 'free' command with direct cgroup data access, making resource queries container-aware, and enhancing the system's ability to prevent OOM errors.

### **Ablator** {[Github Link](#)} | *Ray, PyTorch, Github Action, Docker, Python*

03.2023-07.2023

- Implemented distributed experiment execution with Ray, managed open-source projects, set up CI pipelines via GitHub Actions, oversaw release management and version control, and authored pytest unit tests with 97% coverage.

## **PROJECT EXPERIENCES**

### **Llama3 Emotion Classification** | *LLM, PyTorch, Lora, Flash Attention*

04.2024-07.2024

- Developed a highly accurate emotion text classification model by integrating the Llama3-8b with Lora and FlashAttention techniques, achieving an accuracy of 0.9262, which outperformed models like BERT and RoBERTa.

### **SLinux OS (Similar to Linux 0.11)** | *C, File System, System Call Interface*

06.2023-09.2023

- Developed a functional Unix-like Operating System, providing basic OS functions and ensuring performance after tests.
- Gained hands-on experience with kernel development, file systems, process scheduling, and other core OS concepts.
- Developed modules for memory management, process scheduling, file system management, and process communication.
- Implemented multi-threading and concurrency control to manage and execute multiple processes simultaneously.

### **Job Posting Analytics System** {[Demo](#)} | *Data Mining, Database, Visualization, SpringBoot, ReactJS, Docker*

05.2019-06.2020

- Developed a text mining system for collecting and analyzing recruitment data, which won the "National Level Innovative Excellence Project" award and was adopted by the school.
- Extracted over 10 million recruitment records from job sites using web scraping, with Bloom Filters for deduplication.
- Automated ETL processes using Python (numpy, pandas), and stored data in MongoDB and MySQL.
- Created automated reports, dashboards, and data visualizations with Spring Boot, ReactJS, NodeJS, and AntV, enhancing data readability and visual impact, and integrated Docker with CI/CD pipelines for streamlined project automation.

## **SPECIALIZATIONS & SKILLS**

- **Languages & Tools:** Java, Python, Go, C, SQL, JavaScript, TypeScript, Shell, Bash, Git, Github, Github Action, GitLab
- **Cloud Native:** [Kubernetes](#), [Docker](#), Containerd, Operator, CRD, Helm, Prometheus, Grafana, CI/CD, Github Action, AWS, GCP, GKE, Azure, Linux
- **Backend & Frontend:** Spring, Spring Boot, Spring MVC, Maven, Express, Flask, Gin, Redis, MongoDB, MySQL, Firebase, RocketMQ, JWT, Swagger, Postman, AngularJS, Vue.js, React.js, Bootstrap
- **Machine Learning & Data:** [PyTorch](#), [MLSys](#), Ray, Federated Learning, CNN, GCN, NLP, LLM, Hadoop, Spark