

# Justin He

408-966-5659 | [justinhe@ucla.edu](mailto:justinhe@ucla.edu) | [linkedin.com/in/justinhe24](https://linkedin.com/in/justinhe24) | [github.com/justinHe123](https://github.com/justinHe123) | [justinhe.me](https://justinhe.me)

## EDUCATION

<b>University of California, Los Angeles (UCLA)</b> <i>B.S. in Computer Science, B.S. in Applied Mathematics</i>	Sep. 2020 – Mar. 2024 <i>Los Angeles, CA</i>
<ul style="list-style-type: none"><li>GPA: 4.0/4.0; <i>summa cum laude</i>; Phi Beta Kappa</li><li>Activities: Grader for Programming Languages, Databases; Leadership for Upsilon Pi Epsilon</li></ul>	

## EXPERIENCE

<b>Software Engineer</b> <i>Meta</i>	Apr. 2024 – Present <i>New York, NY</i>
<ul style="list-style-type: none"><li>Facebook Scam &amp; Trust; Tech lead for ML modeling/infrastructure for scam detection on FB Dating Integrity</li><li>Owned end-to-end development and training of production scam detection models, spanning scam trend research, feature engineering, offline evaluation, and online inference; reduced scam view prevalence by 70%</li><li>Re-architected model-serving infrastructure handling 150M+ daily users, introducing caching, batching, and rate limiting to cut direct inference volume by 98%+, and reduce compute spend by \$200k/yr</li><li>Led development of user-facing trust feature, coordinating across 18 cross-functional stakeholders. Achieved 20% adoption in 30 days and scaled into a company-wide trust initiative</li><li>Drove cross-team collaboration with 12 engineers to define and execute a multi-half roadmap to unify FB-wide integrity infrastructure, reducing system fragmentation and improving reliability and operational maintainability</li></ul>	
<b>Software Engineer Intern</b> <i>Citadel Securities</i>	Jun. 2023 – Aug. 2023 <i>New York, NY</i>
<ul style="list-style-type: none"><li>Trading Ecosystem, focused on optimizing trading infrastructure performance</li><li>Built distributed tracing services across order execution infrastructure to enable real-time error detection and correction, cutting investigation time from hours to seconds</li><li>Developed an internal library for streaming Kafka data to q/kdb+, improving data throughput by 1.5x</li><li>Built load-testing infrastructure to simulate 100k+ concurrent orders for trade booking system to benchmark latency and failure modes under stress</li></ul>	
<b>Software Engineer Intern</b> <i>Meta</i>	Jun. 2022 – Sep. 2022 <i>Menlo Park, CA</i>
<ul style="list-style-type: none"><li>Instagram Demand &amp; Efficiency Management, focused on improving Instagram's backend power efficiency</li><li>Led cross-team project to develop a system measuring engagement change per kilowatt used for proposed features</li><li>Built power monitoring dashboard covering 25 services and 500K+ servers, centralizing visibility for energy usage</li><li>Developed algorithm to triage power regressions to launches by correlating launch data with power time-series data</li></ul>	
<b>Software Engineer Intern</b> <i>Fwaygo</i>	Jun. 2021 – Sep. 2021 <i>Los Angeles, CA</i>
<ul style="list-style-type: none"><li>Developed microservices in Go for user/song data processing, report handling, and server-to-client messaging</li><li>Utilized RabbitMQ to facilitate interservice pub/sub communication between Docker container clusters on GKE</li><li>Created GraphQL APIs for user/song queries &amp; mutations and integrated them into a React Native frontend</li></ul>	

## RESEARCH

<b>GNN Integration to Knowledge Graph for Nephrology QA System</b> ( <a href="#">Paper</a> )	2022
<ul style="list-style-type: none"><li>Collaborated with two UCLA PhD students to train novel QA models for joint reasoning across language models and knowledge graph-enhanced GNNs to answer nephrology questions</li><li>Tuned GreaseLM + QA-GNN models through hyperparameter optimization and knowledge graph configurations, achieving 37.2% accuracy on nephrology QA tasks (state-of-the-art at the time)</li><li>Applied mention detection, entity linking, and relation extraction using spaCy to construct a domain-specific knowledge graph and annotated QA dataset</li><li>Extracted and cleaned data from 563 textbook chapters and 814 papers to build high-quality nephrology corpus</li></ul>	

## TECHNICAL SKILLS

**Languages:** Python, C++, SQL, PHP, JavaScript, Go, Java, Bash

**Libraries:** PyTorch, NumPy, Pandas, Matplotlib, React, Node, Express, GraphQL

**Technologies:** Git, Docker, Google Cloud, AWS, Firebase, MySQL, PostgreSQL