

# CHONGBIN (CHARLES) YAO

Seattle, WA ◊ +1(206) 465 8181 ◊ chongbinyao33@gmail.com ◊ [y33.ch](#)

## EDUCATION

---

### University of Illinois Urbana-Champaign

Sept. 2024 - Dec. 2025

Master in Computer Science. GPA: 3.75/4.0

Courses: Paralleled Computing, Deep Learning, Text System, System of Gen AI, CUDA programming

### University of Washington, Seattle

Aug. 2016 - Aug. 2020

Bachelor of Science in Computer Science; *magna cum laude* GPA: 3.83/4.0

## EXPERIENCE

---

### DIFY.AI

Apr. 2025 - Now

*Forward Deployed Engineer/Backend Engineer*

*Seattle, WA*

- **Agentic Workflow Delivery:** Shipped production agentic workflows for enterprise clients on **dify.ai**, integrating **agent guardrails**, **RAG**, **long-term memory**, **sandboxed tool execution**, and **MCP**-based tools and skills.
- **MCP / Tool Plugins:** Converted third-party **REST APIs** into **MCP** tools plugins and **skills.md** for **dify.ai**, expanding the set of capabilities users can compose into their workflows.
- **Per-Workflow OAuth:** Built a **per-workflow OAuth** integration that lets end-users bind their own credentials at runtime, unlocking deeper workflow customization for enterprise tenants.
- **RBAC Service:** Designed and led the implementation of a **role-based access control (RBAC)** service for the **dify.ai** platform.
- **Kubernetes & Helm:** Provisioned **Kubernetes** clusters and authored **Helm charts** for service deployment; diagnosed and resolved production cluster incidents.

### INDEED.COM

Oct 2020 - Aug 2024

*Software engineer*

*Seattle, WA*

- **Full-stack Migration:** Migrated a **Spring-Boot** backend to **Next.js** with **ChakraUI** and swapped **REST** for **GraphQL**; deployed to AWS via **Terraform**, cutting overall latency by 60%.
- **Index Builder Design:** Led the design of a **Spark/Scala** index builder unifying multi-source logs, shortening AB test decision time by 66.7% (2 weeks).
- **Data Processing & Integration:** Led design and implementation of **Spark/MapReduce** pipelines and instrumented native **React Native**, **Kotlin**, and **SwiftUI** apps, enabling user-behavior tracking for 7 sub-teams.
- **User Engagement:** Designed and shipped full-stack features that drove a 300,000-user increase in email opt-ins.

### INDEED.COM

June 2019 - Sept. 2019

*Technical Business Analyst Intern*

*Austin, TX*

- **Application Development:** Built a public-facing **React Native** mobile app end-to-end, driving a 134% lift in targeted user acquisition in the first quarter post-launch.
- **Feature Implementation:** Engineered a **Django REST API** and user profile front-end, reducing server response times by 42%.

## PROJECTS

---

### PERSONAL AI AGENT ASSISTANT

- **The Problem:** Thinking with AI is hard. Few tools help people brainstorm, explore ideas, or build lasting context from their conversations.
- **The Challenge:** Build a desktop app — toggled by a global keyboard shortcut — that lets users think alongside an AI and organize the resulting thoughts through a persistent memory system.
- **The Solution:** A **Tauri** desktop app with a global hotkey toggle, wrapping an **LLM agent** over three tools: a **git-backed markdown wiki** with index/log/entity pages for compounding memory, **tmux sessions** that persist shell state across restarts, and **tiered sandboxing** (`sandbox-exec` / Docker) for safe CLI execution.

### PARALLEL AGENT ORCHESTRATION FRAMEWORK

- **parallel-task-decomposition:** Breaks a goal into a **tasks.md** of file-disjoint, independent, self-contained task blocks, enforcing rules (file-disjointness, either-order correctness, right-sizing) that avoid downstream merge and semantic conflicts.
- **parallel-worktree-agents:** Runs sub-agents on isolated **git worktree** branches, coordinated through the filesystem (shared preamble, append-only broadcasts, git diffs as evidence), with asymmetric knowledge between agents, wave-based dependency scheduling, broadcast verification against diffs, and topological merge convergence.

## TECHNICAL SKILLS

---

**AI related:** Agents orchestration, Skill.md, MCP, Dify, Tool Use / Function Calling, Prompt Caching, Claude / OpenAI APIs, Embeddings, pgvector, Fine-tuning

**Programming Languages:** Java, TypeScript, Scala, Python, SQL, C++, C, C#

**Library and Frameworks:** React, Node.js, Next.js, Spark, PyTorch, ChakraUI, Django, NumPy, PyTorch, React-native, Flutter

**Tools and Software:** Docker, Kubernetes, GraphQL, Terraform, AWS, RESTful API, MongoDB, GraphQL, Unity, UE5