

# CREATE CLOUD NATIVE AGENTS AND EXTENSIONS FOR LLMS

**Vivian Hu**

@alabulei



**Michael Yuan**

@juntao



# Current tech stacks for LLM Apps

- LLM inference – Pytorch + Python
- LLM agent/extension – Langchain + Python



# The Challenges for LLM apps in Cloud Native

The tech stack is too heavy with complex python dependencies.



**Santiago Viquez** ✓  
@santiviquez



The best minds of my generation are thinking about how to install Python.



**Chris Albon** @chrisalbon · 1d

What is "the right way" to install Python on a new M2 MacBook? I assume it isn't the system Python3 right? Maybe Homebrew?

3:42 AM · 7/6/23 from Earth · **744K** Views



**Greg Brockman** ✓  
@gdb



Much of modern ML engineering is making Python not be your bottleneck.

6:55 AM · 7/6/23 from Earth · **244K** Views

# The Challenges for LLM apps in Cloud Native

- The AI apps are not portable due to the diversity of the GPU and CPU devices.
- We're forced to use API server.
- However, API server is not flexible and not performant.
  - multi-models
  - complex RAG
- Even Python is not cross device portable

# How can we solve the problem?



WEBASSEMBLY

# Why Wasm?

- Portable: the compiled Wasm app can be run across different platforms without recompiling.
- Lightweight: The runtime is only 30 MBs and the inference app is only 4 MBs.

# How?

- Wasi-nn proposal
- Support major AI frameworks
  - OpenVINO
  - TensorFlow
  - Pytorch

## wasi-nn

A [Bytecode Alliance](#) project

High-level bindings for writing wasi-nn applications

 CI passing  crates.io v0.6.0  npm v0.3.0

### Introduction

This project provides high-level wasi-nn bindings for Rust and AssemblyScript. The basic idea: write your machine learning application in a high-level language using these bindings, compile it to WebAssembly, and run it in a WebAssembly runtime that supports the [wasi-nn](#) proposal, such as [Wasmtime](#) and [WasmEdge](#).

# Extend LLMs to WASI-NN

WasmEdge-Wasi-NN

- Integrate llama.cpp as a new backend
- llama.cpp is the inference of LLaMA model in pure C/C++



**WasmEdgeRuntime**



# What do we get?

- Write once, run everywhere
- Zero python dependency
- Cloud-ready apps
- Native speed



## Demo #1

Use Wasm as a cross-platform runtime for LLM inference



**Create an LLM web service on a MacBook,  
run it on a NVIDIA device.**



# It's also an OpenAI-compatible API server

- But it's not a simple API server.
- Reuse the ecosystem around OpenAI, like Langchain and flows.network
  - build LLM agent
  - build RAG application
  - build chatbot

# Wasm is the best plugin mechanism for LLMs

- Wasm is a great runtime for lightweight serverless function
- It can manipulate LLM input and output like langchain does,
- but without Python



Reduce human errors

integrate ChatGPT into your automation

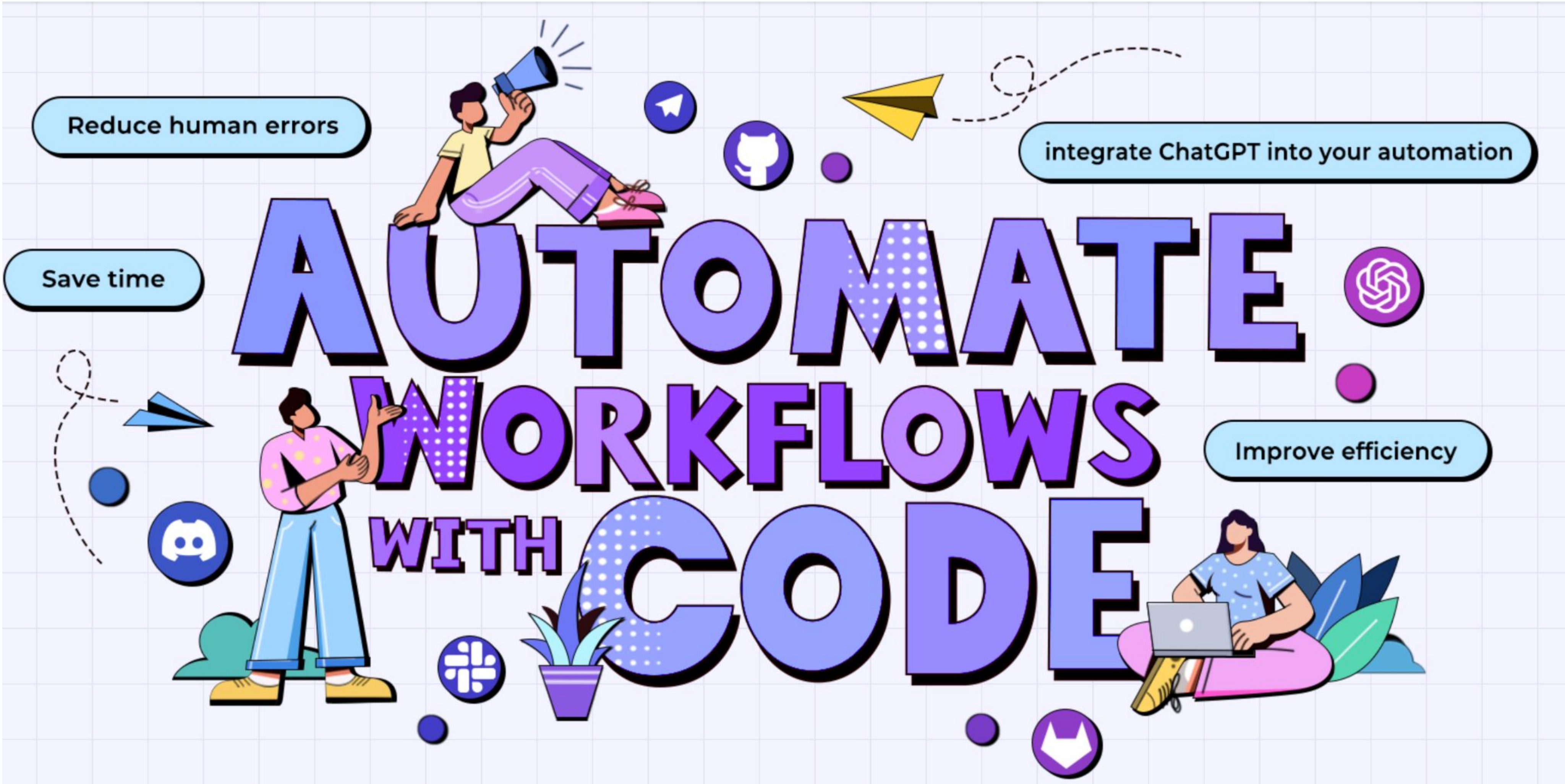
Save time

# AUTOMATE

# WORKFLOWS

# WITH CODE

Improve efficiency



# flows.network

- A serverless platform for LLM powered SaaS workflows
- The function is written in Rust
- WasmEdge is the runtime to run the compiled Wasm binary

# Create a Discord bot with flows.network

- flows.network provides API integration with Discord and WasmEdge-Wasi-nn runtime
- With WasmEdge, you can use any open source LLMs as backend



LLM Service



Discord



## Demo #2

**WASM is the runtime for LLM extensions.**



# LLM agent based on ChatGPT

<https://flows.network/start> (All the function is written in Rust.)

## Code review by ChatGPT/4

Automatically triggered by each new commit in the GitHub Pull Request. Customize the flow function code for your own code review strategy. [Learn more](#)

Summarize & review changes in each commit

Review each changed file in the PR

Summarize the GitHub Issue by sending the trigger world in the issue comment

## Conversation bot for OpenAI Assistant API

Connect your OpenAI application based on the Assistant API to the external world.

Connect your OpenAI applications based on the Assistant API to Telegram

## ChatGPT/4 powered conversation bots

Change to your own prompts, language settings, or even use your own fine-tuned models to customize the bot's behavior.

Telegram

Discord

Slack

GitHub Discussion

## Claude/2 powered conversation bots

Change to your own prompts, language settings, or even use your own fine-tuned models to customize the bot's behavior.

Telegram

## Automate DevRel workflows for Open Source communities

Turn tedious manual workflows into automated flow functions. You can customize each flow based on your own requirements and approaches.

Discord notifications for new Hacker News articles with your keywords, plus a summary provided by ChatGPT.

Slack notifications for new Hacker News articles with your keywords, plus a summary provided by ChatGPT.

Slack notifications for negative sentiment GitHub issues By ChatGPT.

Slack notifications for new Hacker News articles with your keywords.

Slack notification for every 10 GitHub stars

Slack notification for GitHub issues that hasn't been responded to in a certain number of days

Discord notification for the GitHub issues with specified labels

Slack notification for each fork of a GitHub repo

# Roadmap

- Add more LLM-related backends for WASI-NN plugins
  - MLX for Apple chips
  - Intel Extension for Transformers
  - OpenAI's whisper
  - Burn
- Add support for embedding models
- Container management on GPU

# Resources

- WasmEdge: <https://github.com/WasmEdge/WasmEdge/>
- WASI-NN: <https://github.com/bytecodealliance/wasi-nn>
- WasmEdge-WASI-NN:  
<https://github.com/second-state/wasmedge-wasi-nn>
- LlamaEdge: <https://llamaedge.com/>
- Source code: <https://github.com/LlamaEdge/LlamaEdge>
- Flows.network: <https://flows.network/start>

THANKS!

