

赛题三



基于openKylin的人工智能 异构算力调度平台挑战赛



报名网址:

<https://competition.atomgit.com/competitionInfo?id=1e723ae2c04fcd89cae6a8d7ae625749>

实战竞技赛

基于openKylin的人工智能异构算力调度平台挑战赛

赛题背景

随着人工智能技术的飞速发展，AI应用已深入各行各业。为满足海量计算需求，现代单台计算机设备普遍采用CPU、GPU、NPU等组成的异构计算架构。然而，当前开发者面临巨大挑战：硬件异构性导致编程模型复杂，上层应用难以高效、便捷地利用多种计算单元，造成了算力资源的浪费和应用性能的瓶颈。与此同时，国家大力推动开源生态和基础软件建设。openKylin作为国产领先的开源桌面操作系统，亟需构建与之匹配的底层系统软件生态。在此背景下，攻克智能算力调度这一关键共性技术，对于提升国产操作系统在AI时代的技术竞争力、推动国产AI软硬件协同发展、构建自主创新的计算体系具有重大的战略意义和现实需求。

赛题任务

参赛团队需基于openKylin操作系统和国产CPU、GPU、NPU，在单台电脑上设计并实现一个人工智能异构算力调度平台（库或中间件）。该平台应具备以下核心能力：硬件资源感知与管理、任务特征分析与建模、智能调度策略引擎、统一运行时接口、性能评估与展示。

基于openKylin的人工智能异构算力调度平台挑战赛

赛题范围

为了实现基于openKylin操作系统和算力部件（国产CPU、国产GPU、国产NPU），首先需要了解和学习中国算力硬件，其次考虑构思平台架构，最后完成调度平台开发。**可能的解决方案包括：**

- 中国制造算力硬件，算力部件如：海光CPU、兆芯CPU、景嘉微GPU、格兰菲GPU、后摩NPU、登临NPU等
- 终端智慧办公操作系统：openkylin
- 算法模型如：Qwen3等
- . . .

需要解决的问题

- 硬件资源感知与管理：能够感知和管理算力硬件算力、占用率等
- 任务特征分析与建模：能够对输入的AI计算任务（如模型推理、图像处理、矩阵运算等）进行解析和特征提取（如计算密集型、内存密集型、算子类型、数据规模等）。
- 智能调度策略引擎：如何平衡调度各个算力部件
- 统一的运行时API和性能评估展示
- . . .

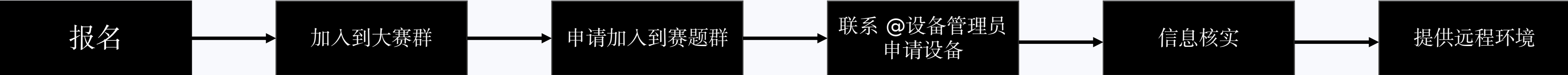
评审细则

1. 平台是否实现了核心功能，单台计算机（国产CPU、GPU、NPU）异构算力调度与可视化监控，稳定运行且无致命错误。 25分
2. 量化数据
计算机单次模型（qwen3-8B开源模型）推理计算所使用的算力需要分布在CPU、GPU、NPU上。 15分
单次执行模型（qwen3-8B开源模型）推理计算所使用到的算力通过算法合理分布在CPU、GPU、NPU上。 15分
3. 调度策略算法（算法讲解）的创新程度、智能性、鲁棒性（算法连续运行24小时）。 15分
4. 代码结构清晰，模块化设计合理，注释完整，符合开源规范，易于他人理解和后续开发。 10分
5. 参赛团队需要提供足够详细的说明文档指导其他开发者运行/使用这个项目 10分
6. 参赛团队项目演示完整度、逻辑表达 10分

最终团队成绩，将在总决赛中的作品答辩环节的综合评分后得出，评出各类奖项。

动手实践（运行环境）

环境申请



扫码加好友，发送“大赛报名”进群

环境检查

配置	OS版本: openKylin 2.0	CPU: 海光3350	GPU: 景嘉微1100	NPU: 后摩M50
查询命令	cat /etc/os-release	lscpu	jm-smi -q	hm_smi -a

技术支持联系方式： 颜老师 15353921091

可能需要的知识

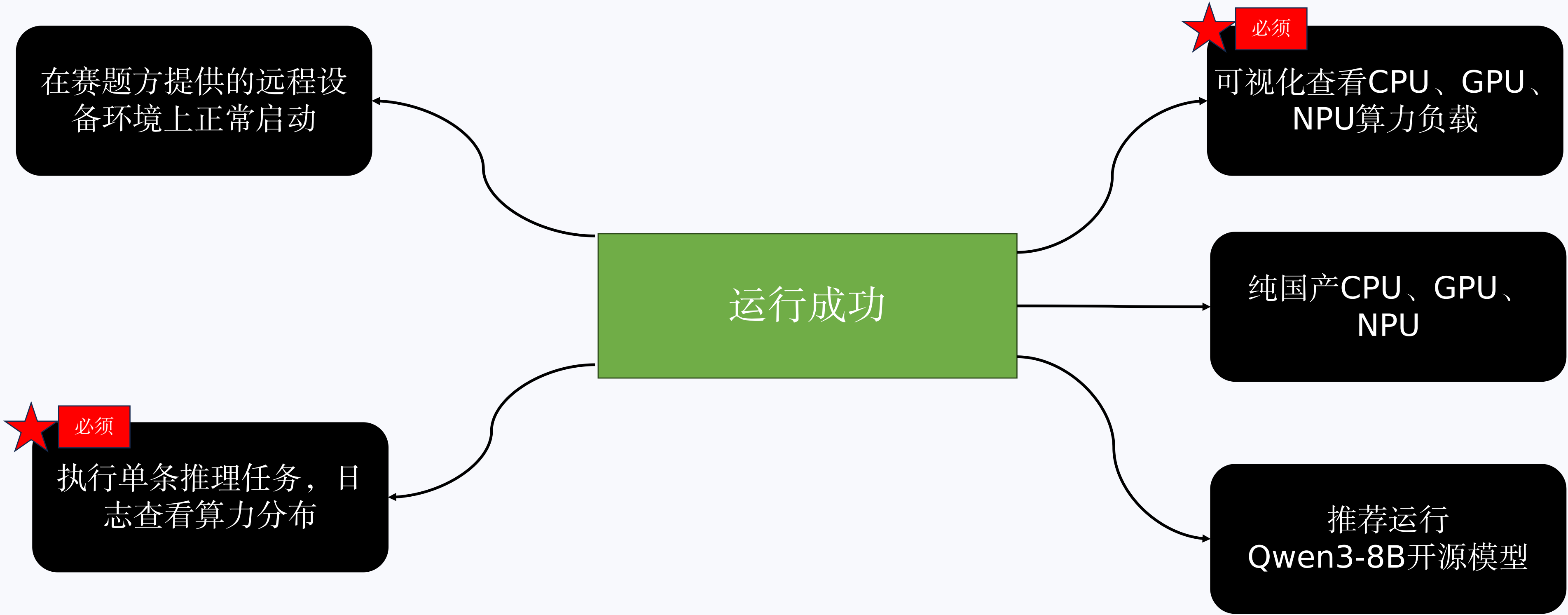
- llama.cpp
- C++
- 算子
- vulkan
- Qwen3-8B
- 计算机原理
- openKylin 2.0

培训、支持

- 培训一：景嘉微显卡使用 预计 9月下旬
- 培训二：后摩NPU卡使用 预计9月下旬
- 培训三：景嘉微算子讲解 预计10月10日
- 培训四：后摩NPU卡 预计10月10日

赛题群技术指导服务时间：工作日9:00 ~ 18:30

景嘉微显卡技术支持：李老师 15520068121
后摩NPU技术支持：谭老师 15580066808
赛题技术支持： 杨老师 15353921091





技术文档

- Llama.cpp: <https://github.com/ggml-org/llama.cpp>
- openKylin 技术文档: <https://docs.openkylin.top/zh/home>
- 景嘉微参考文档:
 - OpenCL: <https://www.khronos.org/opengl>
 - 用户手册: 

MWV207D Linux
Guest1.0软件...

- 后摩参考文档: <https://developer.houmoai.com/doc>



检查景嘉微显卡：

打开终端运行：`jm-smi -q`

输出以下信息表示成功：

```
=====JMSMI LOG=====
Timestamp : Fri Jun 6 17:37:48 2025
Driver Version : 1.0
Attached GPUs : 1
GPU 0000:06:00.0
  Product Name : JMI100
  Product Brand : JMI100
  Display Active : Enabled
  Driver Model
  Current : MESA
  Serial Number : Not Found
  GPU UUID : Not Found
  VBIOS Version : 11.0-20250528.1610
  GPU Virtualization Mode
  Virtualization Mode : None
.....（省略）
```

参数说明请查看用户文档 6.2 节。

检查后摩 NPU：

打开终端运行：`hm_smi -a`
输出以下信息表示成功：

```
-----
sdk build infos
-----
Build_Time : 2025-07-16 11:18:59
HMSW_Version : V0.3.0
HM_SMI_Version : V0.0.2
-----
Mon Jul 28 14:38:08 CST 2025
-----
device0 detail infos
-----
Vendor : Houmo
BDF : 0000:01:00.0
Driver_Version : V0.3.0
Dev : 0
Physical_ID : 0
Cur_BandWidth : 16.0 GT/s-4lane
SN : N/A
Model : N/A
Firmware_Version : V0.3.0

.....（省略）
```

cpu:

$\text{FLOPS} = \text{核数} * \text{单核主频} * \text{CPU 单个周期浮点计算能力}$

$\text{CPU 单个周期浮点计算能力} = \text{FMA(Fused Multiply-Accumulate) 单元数量} * 2 \text{ (乘加运算)} * \text{支持的向量运算位数} / \text{精度位数}$

GPU:

$\text{FLOPS} = \text{单周期计算能力} * \text{核心运行频率} * \text{流处理器数量} * \text{融合乘加次数}$

NPU:

$\text{FLOPS} = \text{单 MAC (Multiply-Accumulate) 的 FLOPS} * \text{频率} * \text{MAC 数量}$

llama.cpp 是 Georgi Gerganov 创建的一个 C/C++ 的模型推理框架，它的主要目标是实现最小化设置和最先进的性能，在广泛的硬件上启用 LLM 推理——无论是在本地还是在云端。

它有以下优点：

- 纯 C/C++ 实现，无任何依赖
- 支持x86架构的AVX、AVX2、AVX512和AMX
- 1.5位、2位、3位、4位、5位、6位和8位整数量化，以实现更快的推理和减少内存使用
- 支持多种后端。
- CPU+GPU混合推理。

llama.cpp 基于 ggml (<https://huggingface.co/blog/introduction-to-ggml>) 开发，有以下几个重要概念：

- backend (执行计算图的接口，支持 OpenCL、Vulkan、CUDA 等)
- buffer (对应 backend 的 内存空间)
- scheduler (调度器，负责分配计算任务)

可编译 llama.cpp 运行 simple 程序，来参考。

2025



开放原子开源基金会
OPENATOM FOUNDATION

openKyLin

欢迎报名



扫一扫 进入赛事页面报名



添加好友，并发送“**大赛报名**”入群交流



www.openKylin.top



contact@openKylin.top