

第八届中国研究生人工智能 创新大赛华为赛题

赛题一

1. 题目名称

可自学习的AOI实时在线AI质检

2. 题目描述

在制造业中，产品缺陷检测通常使用传统计算机视觉算法与人工检测进行，对每种新的缺陷均需要在极少样本（1-2）下开发新算法拦截。本题目旨在开发一套AI驱动的实时质检系统，基于AOI（Automated Optical Inspection）技术，在组装过程中即时监测组件（屏幕、电池、中框等）状态，实现不良事件的早期拦截。该系统需处理视频或图片数据，识别异常模式，并动态优化模型。

本题目将重点解决以下问题：

● 实时视频/图片异常检测：

利用计算机视觉区分正常与异常模式，处理尺寸偏差、缺件少件、逻辑错误（如顺序错误）、色彩变化与常见外观缺陷。需同时覆盖视频输入与图片输入进行检测的场景。对 2500*2500 大小的输入图片，在 2060 及以下GPU上模型运行时间<200ms，可挑战目标：cpu上运行时间<2s。

● 少样本或无样本启动与泛化：

工业场景异常样本获取难度较高，系统需在少样本（few-shot）或无样本（zero-shot）条件下快速部署。解决方案包括迁移学习（从相关任务预训练）、合成数据生成（如GANs模拟缺陷），或在线学习以适应新环境。

● 用户反馈驱动的优化：

当系统误检或漏检时，操作员可提供实时反馈（如标记漏检），系统应能回溯检测逻辑，动态调整模型参数（如通过强化学习或主动学习）。

3. 具体要求

- 1) 参赛者需给出算法模型的可解释性文档，系统阐述方案选型依据，并附上数据/图片等其他可证明观点的论据；
- 2) 选手需利用公开数据集进行模型训练，并自行验证模型的泛化能力；
- 3) 赛题总分由竞赛得分（60%）和专家评分（40%）两部分组成；
- 4) 竞赛得分部分根据选手提交的方案及其在非公开测试集上的运行结果给出。测试集运行前，仅使用 100 张带标注无缺陷图片或视频，与 30 张有缺陷图片或视频进行迁移学习，而后对测试集 1000+ 图片用于测试。本次参赛的最终成绩会综合使用方案完整度（50%）、答案准确率（20%）、检测时间（30%）；
- 5) 专家评分由评委组对选手所提交的方案的新颖性、合理性等进行打分。因此，参赛选手还需要提交模型代码（用于非公开测试集评

估)、模型使用说明文件(用于报告模型方案以及模型在公开测试集上的结果)。鼓励使用小模型。

4. 参考数据集

[DAGM 2007](#)

[MVTec AD](#)

5. 参考文献

[1] J. Zhu, C. Ding, Y. Tian and G. Pang, "Anomaly Heterogeneity Learning for Open-Set Supervised Anomaly Detection," 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2024, pp. 17616–17626, doi: 10.1109/CVPR52733.2024.01668.

6. 咨询专家及联系邮箱

陈鹏飞 – chenpengfei19@huawei.com

陈月莹 – chenyueying@huawei.com

7. 赛事讨论链接

<https://www.chaspark.com/#/races/competitions/1264749317697081344>

赛题二

1. 题目名称

面向多源异构内容的用户意图感知与多目标推荐排序优化

2. 题目描述

在双瀑布流推荐场景（如小红书、快手等）中，系统需要克服服务类（如外卖/酒店）与内容类（如短视频/资讯）卡片间的异构性挑战，实现跨模态协同推荐。用户意图应当有效关联强相关的跨品类内容。例如，当用户意图为“周末放松”时，系统应能够综合推荐“外出旅行”（服务类）与“喜剧电影”（内容类）。

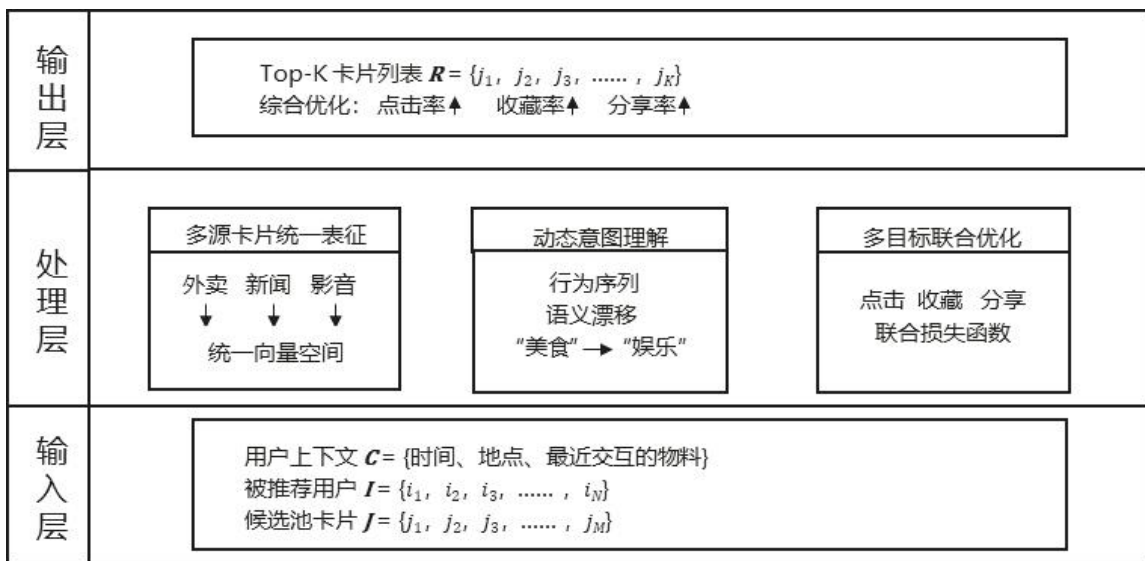
本题目旨在挑战：在统一语义空间下，解决多源异构物料的表征对齐、动态意图捕捉及多目标协同优化问题。参赛者需构建一个精排模型，在满足线上部署约束的前提下，实现用户意图与多源卡片的精准匹配，并平衡点击、收藏、分享等多重业务目标。

本题目将重点解决以下问题：

1. 多模态统一表征：如何将格式、特征差异显著的外卖券、新闻资讯、商品卡片等多源异构物料映射到统一的语义向量空间，进行统一排序的问题。

2. 动态意图演化理解：当用户意图实时变化时（如：美食→电影），如何基于用户行为序列，实现意图转移概率预测的问题。

3. 多目标联合优化：如何在推荐系统中设计多任务学习框架，同时优化点击率、收藏率和分享率三类目标，避免指标间相互冲突的问题。



3. 具体要求

1) 参赛者需要基于开源数据构建推荐精排算法，将多源异构卡片映射到统一语义空间，实现用户意图与卡片的精准对齐，并协同优化点击 (Click)、收藏 (Collect)、分享 (Share) 三类行为，即以用户实时上下文与候选卡片池为输入，输出与用户意图语义对齐的

- 卡片列表；

2) 性能要求：模型参数量参考值 $\leq 800\text{MB}$ ，支持GPU/CPU混合部署；

3) 赛题总分由竞赛得分（60%）和专家评分（40%）两部分组成。专家评分使用华为内部的测试数据进行指标评价和验收，考察算法的泛化性；竞赛得分部分根据选手提交的方案在测试集上的运行结果给出，竞赛得分评分标准计算公式如下：

$$Score = \frac{1}{N} \sum_{i=1}^N \left(0.3 * Hit_{Click}^{(i)} + 0.4 * Hit_{Collect}^{(i)} + 0.3 * Hit_{Share}^{(i)} \right)$$

其中 N 为测试用户数, $Hit_{Click}^{(i)}$, $Hit_{Collect}^{(i)}$, $Hit_{Share}^{(i)}$ 表示在为用户 i 推荐的 Top- K 列表中, 是否发生了对应行为 (发生为 1, 否则为 0)。本题目中, K 设为 20。

4) 专家评分由评委组对选手所提交的方案的新颖性、合理性等进行打分。因此, 参赛选手还需要提交模型代码 (用于非公开测试集评估)、模型的可解释性文档 (用于报告模型方案以及模型在公开测试集上的结果)。

4. 咨询专家及联系邮箱

樊铸锋 - fanzhufeng@huawei.com

5. 参考评估数据集

<https://github.com/RED-Search/Qilin?tab=readme-ov-file>

6. 参考文献

[1] Chen J, Dong Q, Li H, et al. Qilin: A multimodal information retrieval dataset with app-level user sessions[C]//Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2025: 3670-3680.

7. 赛事讨论链接

<https://www.chaspark.com/#/races/competitions/1264750537716064256>

赛题三

1. 题目名称

科研场景下复杂学术查询的智能论文搜索与推荐

2. 题目描述

在科研工作流中，文献检索是最基础也最耗时的环节之一。研究者往往需要基于复杂的、细粒度的研究问题在海量学术文献中找到全面且精准的相关论文集合。这种需求远超传统关键词搜索引擎的能力边界——它要求系统不仅理解查询的语义意图，还需具备多轮检索、引文网络探索、以及跨文献关联推理的能力。

近年来，以大语言模型（LLM）为核心的智能搜索Agent展示了在学术论文搜索领域的巨大潜力。例如，PaSa（ByteDance/北大）通过Crawler+Selector双 Agent架构和强化学习优化，在RealScholarQuery上超越Google+GPT-4o基线37.78%；SPAR通过RefChain查询分解和查询演化，在AutoScholar上达到F1=0.38；Ai2 Paper Finder通过多索引语义检索和多臂老虎机采样策略，在PaperFindingBench上大幅领先通用Agent。

然而，当前方案仍面临以下挑战：

- 查询理解不充分：用户的学术查询往往包含多维度约束（主题、

方法、数据集、时间范围、发表venue等），现有系统难以全面解析

- 检索覆盖率与精确度的平衡：在保证高召回率的同时控制噪声，是论文搜索的核心矛盾
- 搜索论文权威性、时效性、相关性、多样性的权衡
- 根据不同的查询意图，搜索结果综合归纳以及整理，结构化展示

本赛题期望参赛者构建一个端到端的学术论文智能搜索系统，能够针对自然语言描述的复杂学术查询，自动化地完成从查询理解、多维度多策略检索、论文综合排序以及搜索结果归纳整理的全流程。

3. 具体要求

3.1 核心功能要求

(1) 查询理解与分解

- 解析用户自然语言查询中的核心研究意图，识别关键实体、方法论约束、数据/领域限定等多维检索条件
- 对复杂查询进行子查询分解，将宽泛或组合式的学术问题拆解为可独立检索的子问题
- 支持查询改写与扩展，生成更适合检索引擎的查询

(2) 基于大模型的自主搜索策略迭代优化

- 自主规划搜索词进行搜索
- 搜索结果过滤不相干、低质量论文

- 支持迭代式检索策略：根据已找到的相关论文信息，动态调整检索策略和关键词

（3）论文综合排序

- 基于论文标题、摘要及可用全文信息，对候选论文与原始查询进行细粒度相关性评估
- 输出经过排序的最终论文列表，区分高度相关和部分相关的结果

（4）搜索结果归纳整理

- 根据用户查询意图，自主整理归纳搜索结果返回结构化展示（列表、关系图等）

3.2 技术要求

- 参赛者可使用开源或商业LLM（需注明具体模型及版本），鼓励使用开源模型（如Qwen、DeepSeek等）
- 检索后端需对接至少一种学术搜索API（如Semantic Scholar、OpenAlex、PubMed等）
- 方案需具备合理的成本控制意识，在评分中将考虑推理成本（API调用次数、Token消耗量）

4. 评测标准

4.1 评测指标体系

指标	权重	说明
F1 Score	70%	综合衡量精确率与召回率的平衡

运行效率	20%	API调用次数、Token消耗量、端到端延时
回复结果结构化	10%	返回的结果是否使用列表、关系图等

4.2 成绩构成

组成部分	权重	评测方式
竞赛得分	60%	公开测试集（30%）+ 隐藏测试集（30%） 自动评分
专家评分	40%	创新性（15%）、方案落地可行性（15%）、 算法泛化性（10%）

5. 参考数据集

<https://github.com/bytedance/pasa>

<https://github.com/allenai/asta-bench>

6. 参考文献

[1] He Y, Huang G, Feng P, et al. PaSa: An LLM Agent for Comprehensive Academic Paper Search. ACL 2025.

arXiv:2501.10120.

[2] Ajith A, Xia M, Chevalier A, et al. LitSearch: A Retrieval Benchmark for Scientific Literature Search. EMNLP 2024.

arXiv:2407.18940.

[3] Feldman S, et al. AstaBench: Rigorous Benchmarking of AI

Agents with a Scientific Research Suite. arXiv:2510.21652, 2025.

[4] Shi X, Li Y, Kou Q, et al. SPAR: Scholar Paper Retrieval with LLM-based Agents for Enhanced Academic Search. arXiv:2507.15245, 2025.

[5] Skarlinski M, et al. Language Agents for Answering Questions from Scientific Literature. NeurIPS 2024.

[6] Khattab O, Santhanam K, Li X, et al. Demonstrate-Search-Predict: Composing Retrieval and Language Models for Knowledge-Intensive NLP. arXiv:2212.14024, 2022.

[7] Feng P, He Y, Huang G, et al. AGILE: A Novel Framework of LLM Agents. NeurIPS 2024.

[8] Muennighoff N, et al. GritLM: Generative Representational Instruction Tuning. arXiv:2402.09906, 2024.

[9] Press O, Zhang M, Min S, et al. Measuring and Narrowing the Compositionality Gap in Language Models. arXiv:2210.03350, 2022.

[10] Lee D, Sohn S S, Lee B, et al. Domain-aligned LLM Framework for Trustworthy Scientific Q/A via Query Reformulation RAG. ChemRxiv, 2025.

7. 参考系统

系统	核心方法	性能参考
PaSa-7B	Crawler+Selector 双Agent + RL训练	RealScholarQuery上 Recall@20超Google+GPT-4o 37.78%
SPAR	RefChain查询分解 + 查询演化 + 多 Agent	AutoScholar上F1=0.38, 超 PaSa 56%
Ai2 Paper Finder	多索引语义检索 + LLM改写 + 引文追 踪	PaperFindingBench上得分 超ReAct Agent一倍以上
PaperQA2	全文检索 + LLM QA	LitQA2上表现优异

8. 咨询专家及联系邮箱

胡文翔 - huwenxiang3@huawei.com

9. 赛事讨论链接

<https://www.chaspark.com/#/races/competitions/1264753432170905600>

赛题四

1. 题目名称

基于扩散模型的实时视频生成算法时域一致性优化

2. 题目描述

基于扩散模型的图像与视频生成技术正处于高速发展阶段，该技术的普及使得普通用户亦能借助相关应用实现个性化创作。随着技术的迭代升级，视频生成质量在时域连续性、画面分辨率等关键指标上得到显著提升，部分生成结果已达到视觉上的以假乱真水平。然而，当前主流的文生视频、图生视频及视频生视频应用多采用离线处理模式，存在显著的延迟问题，难以满足手机实时录像风格迁移、网络直播、虚拟主播等对实时性有严格要求的场景需求，因此，实时视频生成技术已成为当前计算机视觉领域的研究热点之一。

近年来，已有相关研究成果能够实现实时视频帧率（如 30fps、60fps）下的视频生成任务，但此类技术往往以牺牲模型性能为代价，具体表现为模型参数量缩减与扩散步数减少。这一 trade-off 不可避免地导致视频生成质量下降，其中，时域一致性作为人眼视觉感知最敏感的指标，其劣化问题尤为突出——诸如生成视频中主体目标、背景环境的突发跳变等现象，严重影响视频生成的有效性与可用性。基于此，本赛题提出以现有实时视频生成算法为基础，针对

视频时域一致性开展针对性优化研究，同时严格保障算法的实时性要求，旨在解决实时性与生成质量之间的核心矛盾。

3. 具体要求

1) 任务:

给定一组原始视频和部分简单提示词，参赛者设计和实现基于 Diffusion 的算法模型，推理输出根据提示词生成的视频序列

2) 参考算法和数据集:

a) 训练和预训练模型：可以参考以下几个代码仓库 [1]~[4]

b) 数据集：可以参考以下公开数据集：[5]~[7]

3) 实时性约束：要求算法生成视频流时每个单帧图像的延时至少小于 $1/30s$ 。生成分辨率参考：512x512。

4) 参赛者需撰写相关文档，说明所提出算法/方案的有效性和先进性，进行数据集收集方法、数据集增强、模型设计思路等内容的详细解释。

[1] <https://github.com/cumulo-autumn/StreamDiffusion/tree/main>

[2] <https://github.com/guoyww/AnimateDiff>

[3] <https://github.com/williamyang1991/Rerender-A-Video>

[4] <https://github.com/baaivision/vid2vid-zero>

[5] <https://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html>

[6] <https://davischallenge.org/>

[7] <https://opendatalab.com/OpenDataLab/WebVid-2M>

4. 评分标准

1) 感知损失指标 LPIPS, 占 30%

2) 时域一致性指标: 占 50%, 计算方式为:

$$L = \left\| (Y^{t+1} - Y^t) - (X^{t+1} - X^t) \right\|_1$$

其中 Y 为扩散模型输出视频序列, t 为对应帧。X 为源域视频序列。

3) 主观视觉效果评估: 占 20%

5. 咨询专家及邮箱

王学诚-wangxuecheng8@huawei.com

6. 赛题讨论链接

<https://www.chaspark.com/#/races/competitions/1264754810932436992>

赛题五

1. 题目名称

智能终端多模态输入感知与识别及个性化应答

2. 题目描述

随着智能终端（如智能手机、AI PC、智能音箱、车载助手等）的普及，用户与设备的交互方式日趋多元，不再局限于单一的文字或语音指令。在真实场景中，用户往往通过**语音+手势+面部表情+环境上下文**等多种模态的组合来表达意图，例如用户一边看向屏幕上的某个联系人，一边说出“给他发消息”。这种多模态融合的输入方式对终端设备的感知与理解能力提出了更高要求。

然而，当前端侧智能体在多模态输入感知与识别方面仍面临诸多挑战：

- 1) **模态异步与对齐**：不同模态的输入（如语音指令与手势动作）可能在不同时间点发生，需要在时间轴上进行精准对齐，才能正确理解用户意图；
- 2) **多模态冲突与歧义**：当不同模态传达的信息不一致时（如用户嘴上说“不要”，但点头表示同意），系统需要判断真实意图；
- 3) **个性化应答**：系统需要根据用户的历史交互习惯、偏好设置、当前上下文生成个性化、自然且符合情境的应答内容，而非千篇一律的回复；
- 4) **端侧资源受限**：终端设备算力、内存、功耗有限，要求模型轻量化，同时保持高实时性和高精度；
- 5) **隐私保护**：多模态数据（如人脸图像、语音特征）涉及高度隐私，系统需支持端侧闭环处理，避免数据上传云端。

本题目旨在设计一个能够运行于智能终端（如手机、PC）的多模态输入感知与个性化应答系统，通过融合视觉、语音、文本等多种模态信息，准确理解用户意图，并生成自然、个性化的应答内容，实现高效、隐私、低延时的端侧智能交互。

3. 具体要求

1) 系统功能要求:

- **多模态输入感知与对齐:** 支持至少两种模态输入（如语音+视觉、语音+文本、语音+手势等），设计方法实现多模态信息在时间维度上的对齐与融合；
- **意图理解与场景建模:** 结合上下文（如当前界面、历史对话）与多模态输入，准确识别用户当前意图，并进行场景建模（如“消息发送”、“音乐播放”、“日程查询”等）；
- **个性化应答生成:** 基于用户画像（如常用应用、使用习惯、偏好设置）生成个性化应答内容，应答形式可以是文本、语音或图形界面反馈；
- **系统部署与性能:** 交互系统需支持在端侧（如手机、PC）部署，但推理大模型允许部署在服务器或调用公开 API，参数量不超过 14B，允许额外使用 3 个 3B 小模型进行数据处理，单轮对话响应延时<5s。

2) 文档要求:

- 参赛者需提交方案设计文档，系统阐述多模态对齐、意图理解、个性化应答等关键模块的设计原理、模型选型依据及技术合理性；

3) 数据与训练策略:

- 参赛者可利用公开数据集（如 CMU-MOSEI、IEMOCAP、中文多模态对话数据集等）进行模型训练或数据增强；
- 允许结合少量人工标注数据（如不超过 500 条）优化模型在特定场景下的表现，但需在报告中明确标注数据来源与标注策略；
- 鼓励使用合成数据、迁移学习、自监督学习等手段提升模型泛化能力。

4) 提交内容与评分构成:

- 赛题总分由**竞赛得分（50%）**和**专家评分（50%）**组成；
- 竞赛得分基于公开测试集自动评测结果，考察系统在标准场景下的性能；
- 专家评分由评委组对选手提交的方案新颖性、合理性、端侧部署可行性等进行综合评定，同时使用华为内部的多模态交互测试数据集进行泛化性评估；
- 参赛选手需提交：
 - 方案代码（包含推理脚本、模型权重或模型结构说明，支持在端侧运行）；
 - 使用说明文件（详细描述系统架构、模块功能、运行流程、数据处理方式）；
 - 可解释性文档（包括关键模块设计思路、调优策略、实验分析等）；
 - 在公开测试集上的运行结果报告。

4. 咨询专家及联系邮箱

彭剑威- pengjianwei2@huawei.com

5. 赛题讨论链接

<https://www.chaspark.com/#/races/competitions/1264756163677831168>

赛题六

1. 题目名称

基于CANN的分布式通信算法设计与优化

2. 题目背景

近年来,大语言模型与多模态基础模型的参数量呈指数级扩张,从百亿、千亿迈向万亿级别,其复杂的深度神经网络架构与稀疏混合专家(MoE)设计进一步加剧了计算与存储压力。单卡算力与显存已远不足以支撑大规模训练与推理,必须依托数据并行、张量并行、流水线并行及专家并行等多维分布式策略,将模型参数、梯度与激活值拆解至数百乃至数千张 GPU 组成的高性能集群中协同执行。在此过程中,跨卡参数分发、梯度聚合、张量同步与专家路由等集合通信操作的开销急剧膨胀,通信耗时在单轮迭代中的占比从早期小模型的不足 10%,攀升至千亿级模型训练的 30% - 50%,在 MoE 架构与千卡级集群场景下更突破 60%。随着 GPU 计算性能持续提升、

网络带宽增长相对滞后，通信瓶颈已从辅助性开销转变为制约系统扩展性、算力利用率与训练收敛速度的核心壁垒。传统通信库与拓扑策略难以适配大模型高并发、高带宽、低延迟的通信需求，导致大量计算资源长期处于等待状态，集群有效算力无法充分释放。在此背景下，面向大模型分布式场景的通信机制创新、传输算法优化、拓扑感知调度与计算 — 通信重叠技术，成为突破性能天花板、支撑超大规模模型高效训练与部署的关键路径，亦是当前 AI 基础设施领域亟待攻克的前沿课题。

3. 核心目标

面向昇腾NPU集群(单机多卡/多机多卡),参赛团队需依托智能Agent技术，自主完成HCCL集合通信算法

(AllReduce/AllGather/ReduceScatter等)的全流程设计与代码开发，实现高性能、高可靠、低延迟的分布式通信能力，深度适配昇腾兼容的高速硬件链路(HCCS/RoCE/PCIe)，突破大模型分布式训练的通信瓶颈，提升集群算力利用率与线性加速比。本次赛题要求参赛团队提交完整的Agent技术成果，包括Agent能力清单(Skills)、Prompt工程设计方案及可独立运行的Agent工程，赛事验收策略(性能、正确性等评判标准)保持不变。考虑到参赛学生获取昇腾实体硬件难度较高，本赛题允许参赛团队自行通过模拟器完成成果验证，模拟器需能精准模拟昇腾NPU集群的通信特性、拓扑结构及硬件参数，确保验证结果真实可追溯，提交作品时需同步附带模拟器配置说明、

验证流程及完整日志。

本赛题旨在引导参赛团队依托智能Agent技术,基于昇腾硬件核心特性(高速片间互联、计算核、片上缓存),完成全部算法设计与代码开发,实现“算法-拓扑-硬件”深度协同,有效解决分布式训练“通信墙”及算法开发效率低的行业难题,同时提交Agent能力清单(Skills)、Prompt工程设计方案及完整Agent工程。

4. 任务范围

1) 通信原语与算法要求

参赛团队需覆盖昇腾HCCL核心集合通信原语(至少3种),且所有算法设计及对应代码开发工作,需通过智能Agent完成,尽量避免人工直接编写核心逻辑:

- AllReduce (分布式训练梯度同步,核心)
- AllGather (模型并行参数聚合)
- ReduceScatter (数据并行分片归约)
- Broadcast (参数初始化广播)
 - AlltoAll (序列并行/专家并行全局交换)

2) 硬件与拓扑适配场景

- 单机多卡: Full Mesh 全互联,要求 Agent 生成的算法需深度适配全互联特性,充分发挥硬件带宽优势。
- 异构集群: 混合昇腾 910A2/910A3、不同代际 NPU、非对称带宽链路,要求 Agent 具备异构环境感知能力,生成可适配不同

硬件配置的算法代码。

- 极限场景：小数据块（ $\leq 64\text{KB}$ ）、大数据块（ $\geq 1\text{GB}$ ）、动态拓扑（节点增减），要求 Agent 生成的算法具备场景自适应能力，确保不同场景下的通信性能与稳定性。

3) 关键技术要求

技术维度	昇腾集群通信专属要求
硬件感知	自动探测高速硬件链路（HCCS/RoCE/PCIe）的带宽、延迟、误码率等关键参数；精准感知NPU NUMA拓扑、HBM分区、片上缓存（UB）容量等硬件信息；要求Agent具备完善的硬件特性自动探测与建模能力，能够基于探测结果生成适配硬件的算法代码，确保算法与硬件特性深度匹配。
算法创新	突破标准Ring/Mesh/HD算法局限，设计自适应拓扑算法（如NHR非均匀环、分块Mesh、动态Butterfly）、稀疏通信算法、通算融合算法；所有算法设计及代码开发需通过Agent完成，要求Agent具备算法创新生成与迭代优化能力。
软硬协同	利用硬件级通信原语、随路归约、计算通信并发

	调度，实现“零CPU介入、低延迟、高吞吐”；要求Agent生成的代码充分体现软硬协同逻辑，深度适配昇腾硬件架构特性。
可靠性	支持链路故障自愈、数据校验、超时重传、流量控制，避免千卡集群“单点故障雪崩”；要求Agent生成的代码包含完整的可靠性机制，确保集群通信的稳定运行。
可扩展性	算法复杂度控制在 $O(N)$ ~ $O(N\log N)$ ，支持8→1024卡线性扩展，加速比 $\geq 90\%$ ；要求Agent生成的算法具备良好的可扩展性，适配不同规模的昇腾NPU集群。
精度保障	支持FP16/BF16/FP32混合精度通信，误差 $\leq 1e-6$ ，无精度溢出/下溢；要求Agent生成的代码严格满足精度要求，确保通信过程中的计算正确性。
Agent技术要求	依托Agent完成全部算法设计与代码开发，参赛团队需提交Agent能力清单、Prompt工程设计方案及可独立运行的完整Agent工程，确保评委可复现Agent生成代码与算法的全过程。

4) 开发约束

- 基于昇腾 CANN 8.0 或之后的版本、只能基于 HCOMM 开源仓（Gitee: ascend/cann-hcomm）提供的接口进行开发，交付件参考 cann-hccl。全部核心算法及代码需通过 Agent 完成，严禁人工直接编写核心逻辑。
- 代码需采用 C/C++语言开发，严格遵循 HCCL 插件接口（hcclCommInit/hcclAllReduce 等），确保 Agent 生成的代码规范、可编译、可运行。
- 严禁修改昇腾硬件底层驱动、高速链路固件，Agent 生成的代码需严格遵循此约束，仅可基于昇腾官方提供的 API 进行开发。

所有成果可通过自行搭建的模拟器完成验证，模拟器需能精准模拟昇腾NPU集群的通信特性、拓扑结构及硬件参数；若具备实体硬件条件，优先采用昇腾910A2/910A3集群实机验证。验证用例可由Agent辅助生成，确保代码与算法的可行性、稳定性及性能达标，提交作品时需附带模拟器（或实机）的配置说明、验证流程及完整日志，确保验证过程可追溯。

5) 技术实现路径

拓扑探测与建模: 调用HCOMM接口hcclGetTopology获取NPU物理连接（高速链路）信息，构建加权有向图，完成最小生成树、最优通信路径、链路拥塞预测的算法建模及相关代码开发，上述全过程需通

过Agent独立完成，确保建模逻辑与代码生成的一致性。

1. 算法库设计：针对不同数据块大小，由 Agent 自动生成适配的通信算法——小数据采用 Butterfly（递归加倍）、PairWise（低延迟）算法；中数据采用 NHR（非均匀环）、动态分块 Ring（适配昇腾片上缓存）算法；大数据采用 Mesh（单机全互联）、分层 Fat-Tree（多机）算法；稀疏场景采用梯度稀疏化通信（仅同步非零值）、量化压缩（BF16→INT8）算法，明确 Agent 生成逻辑与迭代过程。

昇腾硬件优化：由Agent生成昇腾硬件优化相关代码，充分利用高速链路随路归约（不占用计算核）、片上缓存（UB）复用（通信数据直接驻留UB，实现零DDR拷贝）、计算通信重叠（NPU计算时后台并发通信，隐藏100%通信延迟）等核心特性，确保代码深度适配昇腾硬件架构，最大化发挥硬件性能。

2. 可靠性机制：由 Agent 生成完整的可靠性机制代码，包含链路健康检测（周期性心跳、带宽/误码率监控）、动态路由切换（主链路故障时，100ms 内切换备用路径）、数据校验（CRC32+奇偶校验，重传率 $\leq 0.1\%$ ），确保集群通信的高可靠性。

Profiling与调优：利用昇腾msprof工具或模拟器自带的性能采集工具，采集带宽、延迟、缓存命中率、链路利用率等关键性能指标；基于BERT/LLaMA 2大模型完成端到端验证，对比吞吐量、NPU利用率、训练耗时等核心指标，形成性能分析报告。调优相关代码由Agent迭代生成，同步提交Agent调优相关的能力清单与Prompt设计，赛事

验收策略（性能、正确性等评判标准）保持不变。

3. Agent 工程实现：搭建完整的 Agent 工程架构，明确 Agent 核心能力模块（如硬件感知、算法生成、代码编译适配、迭代调优等），设计科学合理的 Prompt 体系引导 Agent 生成符合要求的算法与代码，确保 Agent 可独立完成全部核心开发任务，并提交可独立运行的完整 Agent 工程文件。

5. 参赛作品要求

1. 代码包：包含 HCCL 算法插件（.so 动态库）、头文件、编译脚本（CMake）、测试用例（单机 8 卡/多机 64 卡）、压测脚本、故障注入工具；所有代码需通过 Agent 生成，需附带 Agent 运行日志、Prompt 调用记录等，完整呈现代码生成过程。

技术文档：包含算法设计说明书（拓扑结构、算法流程、复杂度分析、昇腾硬件适配要点）、性能测试报告（带宽/延迟/加速比、Profiling 可视化分析、与基准算法对比数据）、可靠性报告（72 小时高并发压测结果、故障自愈能力验证、重传率统计）；同时需专项说明 Agent 相关技术，明确 Agent 能力清单、Prompt 设计思路与优化过程、Agent 工程架构及运行逻辑。若采用模拟器验证，需额外补充模拟器配置说明、验证流程、性能模拟参数及完整验证日志，确保验证结果可追溯、可复现，与实机验证标准保持一致。

演示材料：5 分钟演示视频，内容涵盖算法原理、硬件协同机制、性

能对比结果、大模型端到端验证效果；需额外展示Agent生成全过程（含Prompt输入、Agent算法输出、代码生成及验证的完整流程）。若采用模拟器验证，需同步展示模拟器验证操作流程及关键结果，直观呈现Agent的开发能力与成果有效性，清晰说明模拟器与实机环境的适配逻辑。

2. Agent 专项输出物：需单独提交完整的 Agent 工程（含运行环境配置说明、依赖库清单、执行脚本）、Agent 能力清单、Prompt 文件（含引导 Agent 生成算法、代码、测试用例的完整提示词体系），确保评委可复现 Agent 生成代码与算法的全过程。

6. 典型挑战与突破方向

昇腾高速链路深度适配：深入掌握高速总线全互联特性，设计无冲突Mesh调度算法，突破传统Ring算法带宽上限；同时优化Agent的硬件感知能力，确保Agent生成的算法代码可充分适配高速链路，精准匹配硬件特性，最大化发挥硬件极致性能。

- 跨节点 RoCE 优化：解决 Fat-Tree 拓扑下跨节点通信的拥塞、长延迟、数据乱序等问题，实现多机集群线性加速；提升 Agent 生成跨节点通信代码的能力，优化 Agent 对 RoCEv2 链路特性的适配逻辑。
- 动态拓扑自适应：实现集群节点热插拔时算法的自动重构，确保训练过程无中断；优化 Agent 的动态拓扑感知与算法自适应生成能力，使 Agent 可根据拓扑变化实时迭代代码，保障通信稳定性。

- 通算融合极致优化：实现通信与昇腾计算核计算的流水并发，使硬件利用率逼近 100%；通过优化 Prompt 设计，引导 Agent 生成具备计算通信重叠逻辑的代码，充分挖掘昇腾硬件的并行计算潜力。
- 超低精度通信：实现 FP8/INT4 量化与无损压缩技术融合，使通信带宽提升 2^4 倍且保证精度无损；优化 Agent 对量化压缩算法的生成能力，确保 Agent 生成的代码可实现精度与性能的平衡。
- Agent 能力优化：突破 Agent 在算法创新、硬件适配、代码规范生成等方面的技术瓶颈，设计高效的 Prompt 体系，完善 Agent 核心能力模块，确保 Agent 可独立完成全部符合要求的算法与代码开发，且生成过程可复现、可追溯。

模拟器验证适配：搭建可精准模拟昇腾NPU集群特性的模拟器，重点解决模拟器与实机环境的性能差异问题，确保验证结果能真实反映算法实际性能；同时通过Agent优化代码适配模拟器环境，提升验证效率与准确性，确保模拟器验证结果与实机验证标准具备一致性。

113-4677903-1009021

7. 评判标准

注：本赛题验收策略保持不变，核心围绕参赛作品的功能正确性、通信性能、算法创新性、工程化水平等维度开展评判，同时新增对 Agent 专项输出物（能力清单、Prompt 工程、Agent 工程）的完整性、合理性及可复现性评判；考虑到硬件获取难度，采用模拟器验证的

作品，其验证结果将作为性能评判的重要参考，与实机验证结果同等对待，具体评判标准如下：

通信原语正确性：正确实现

AllReduce/AllGather/ReduceScatter/Broadcast等至少3种核心集合通信；输出结果与昇腾原生HCCL基准结果数值一致，无精度漂移、无NaN/溢出；支持FP32/BF16/FP16混合精度；核心算法与代码需通过Agent生成，且可通过Agent完整复现生成过程，生成逻辑清晰、可追溯；采用模拟器验证的作品，需提交完整验证日志，确保功能正确性可复现，与实机验证要求保持统一。

昇腾环境兼容性：基于昇腾CANN与HCCL标准接口开发，不依赖私有API；可在昇腾910B/910C等NPU或对应模拟器上正常编译、运行、链接；支持单机8卡、多机多卡集群部署；Agent生成的代码需完全满足上述兼容性要求，无适配性问题。

8. 咨询专家及联系邮箱：

石楠翔 - shinanxiang@huawei.com

苏建加 - sujianjia@huawei.com

9. 赛题讨论链接

<https://www.chaspark.com/#/races/competitions/1264757215080792064>

赛题七

1. 题目名称

面向低算力端侧平台基于视觉的实时跌倒检测

2. 题目描述

在居家养老、医疗照护等场景中，跌倒检测的准确性和实时性对挽救生命至关重要。伴随智能安防设备的普及，当前的智能化看护产品聚焦于在端侧（边缘计算设备）构建视觉智能体，为用户带来安全保障的同时，实现视频流等隐私数据在端侧闭环，避免云端传输带来的隐私泄露风险和网络延迟。然而，端侧设备受限于计算资源、存储空间和功耗，只能运行轻量级的小尺寸视觉模型，难以准确捕捉和理解复杂的三维人体动作语义。在实际的纯视觉跌倒检测场景中，传统方案存在以下痛点：

1) **动作语义混淆：**人体跌倒过程复杂多变，与日常动作（如快速坐下、弯腰捡物、躺下休息等）在视觉表征上极为相似，导致轻量级模型误报频发。

2) **算力与效率瓶颈：**端侧算力低、存储资源少，高精度、高复杂度的视频动作分析网络（如庞大的 3D-CNN 或 Video Transformer），推理时延高，无法满足跌倒检测的实时性要求。

3) **场景长尾问题：**在傍晚微光、夜间红外、人体存在遮挡、摄像机视角变化等长尾恶劣场景下，轻量级算法特征提取能力不足，

检测精度下降，造成的漏报和误报。

为解决以上痛点，本题目期望基于小尺寸的轻量级视觉模型，搭配高效的后处理或其他轻量算法，在计算资源受限的端侧设备上，完成人体跌倒行为的精准语义理解与实时报警。

本题目将重点解决以下核心挑战：

1) **跌倒检测行为的准确识别**：精准识别视频流中的人体跌倒，能够将真实跌倒行为与日常行为（坐、弯腰、站立行走，蹲等）区分开，在保证高召回率的同时，具有较低的误报。

2) **低算力下的高效推理构建**：在保证检测精度的前提下，探索模型轻量化架构设计或网络压缩技术，将算法方案在端侧的单帧端到端推理耗时控制在数十毫秒级。

3) **环境干扰的泛化性优化**：针对长尾问题，通过高效的数据增强、域适应或轻量级时空特征融合策略，提升算法在复杂光照、不同机位视角以及遮挡等小样本/极端场景下的泛化与识别能力。

4. 具体要求

1) 端到端纯视觉解决方案设计：构建端到端纯视觉跌倒检测算法解决方案，不得依赖其他深度图、穿戴设备输入信息等，需要支持红外图像的跌倒检测。

2) 性能要求：

- 模型总参数 $\leq 20\text{M}$ ，总参数量 $\text{fp32} \leq 80\text{MB}$ ，支持端侧设备（海思、瑞芯微等 NPU）部署；

- 相机图像输入分辨率为 1080P 以上，算法输入分辨率根据整体解决方案自行定义；
- 端侧推理耗时 $\leq 100\text{ms}$ ，推理时 NPU 存储占用 $\leq 20\text{MB}$ ；
- 训练所需存储、算力资源自备不做要求，可利用现有真实数据样本或结合其他开源数据集，进行数据增强、合成等方式来保证模型的泛化能力；
- 测试基于视频片段进行指标的评测计算，每个片段可能包含一个或者多个跌倒场景，生活场景等。

3) 竞赛得分部分根据选手提交的整体端到端算法解决方案在测试集（包括公开测试集和非公开测试集）上的运行结果得出，会提供少量非公开测试集场景数据供参考；竞赛得分最终成绩由模型检测的 90 和 95 召回率情况下的平均精准率 MAP（50%）、整体模型参数量（25%）、模型推理耗时（25%）综合得出：

跌倒 True Positive (TP)：真实跌倒并且检测为跌倒的

跌倒 False Positive (FP)：日常行为误检测为跌倒的

跌倒 False Negative (FN)：真实跌倒但是没检测到跌倒，漏检的

召回率 (Recall)：
$$R = \frac{TP}{TP + FN}$$

精准率 (Precision)：
$$P = \frac{TP}{TP + FP}$$

定义 P_{90} 是指在召回率为 90% 时的精准率， P_{95} 是在召回率为 95% 时的精准率，算法的整体得分为：

$$MAP = \left(\frac{P_{90} + P_{95}}{2} \right) * 100$$

模型参数量基线得分为 25 分，少于约束根据实际参数量可以获得额外的得分最高为 10 分，超过约束会进行扣分，最大允许超过约束规定的 1 倍，也就是参数量上线 40M，超过则为 0 分：

$$S_{model} = \begin{cases} 25 + 10 * \left(\frac{20 - Size_{model}}{20} \right), & Size_{model} \leq 20 \\ 25 * \left(1 - \frac{Size_{model} - 20}{20} \right)^2, & 20 < Size_{model} \leq 40 \\ 0, & Size_{model} > 40 \end{cases}$$

推理耗时同模型参数量一样，基线得分为 25 分，少于约束根据实际参推理时间可以获得额外的得分最高为 10 分，超过约束会进行扣分，最大允许超过约束规定的 1 倍，也就是推理时间 200ms，超过则为 0 分：

$$S_{time} = \begin{cases} 25 + 10 * \left(\frac{200 - Time_{inference}}{200} \right), & Time_{inference} \leq 100 \\ 25 * \left(1 - \frac{Time_{inference} - 200}{200} \right)^2, & 100 < Time_{inference} \leq 200 \\ 0, & Time_{inference} > 200 \end{cases}$$

最终竞赛得分： $S_{final} = 0.5 * MAP + 0.25 * S_{model} + 0.25 * S_{time}$

4) 赛题总分由竞赛得分（70%）和专家评分（30%）两部分组成。专家评分由评委组对选手所提交的方案的新颖性、合理性等进行打分。因此，参赛者需要提交端到端算法解决方案文档，详细描述整体方

案的设计思路，各模块模型选型和设计，给出可行性、先进性以及相对于业界的优化点分析，以及模型代码(用于非公开测试集评估)、模型使用说明文件(用于报告模型方案以及模型在公开测试集上的结果)。

4. 参考数据集

<https://huggingface.co/datasets/simplexsigil2/omnifall>

<https://www.kaggle.com/datasets/payutch/fall-video-dataset>

5. 参考文献

[1] LFD-YOLO: a lightweight fall detection network with enhanced feature extraction and fusion

[2] BMR-YOLO: A deep learning approach for fall detection in complex environments

[3] YOLO-fall: a YOLO-based fall detection model with high precision, shrunk size, and low latency

6. 咨询专家及联系邮箱

何建忠 - jianzhong.he@huawei.com

7. 赛题讨论链接

<https://www.chaspark.com/#/races/competitions/1264757525464276992>