openKylin

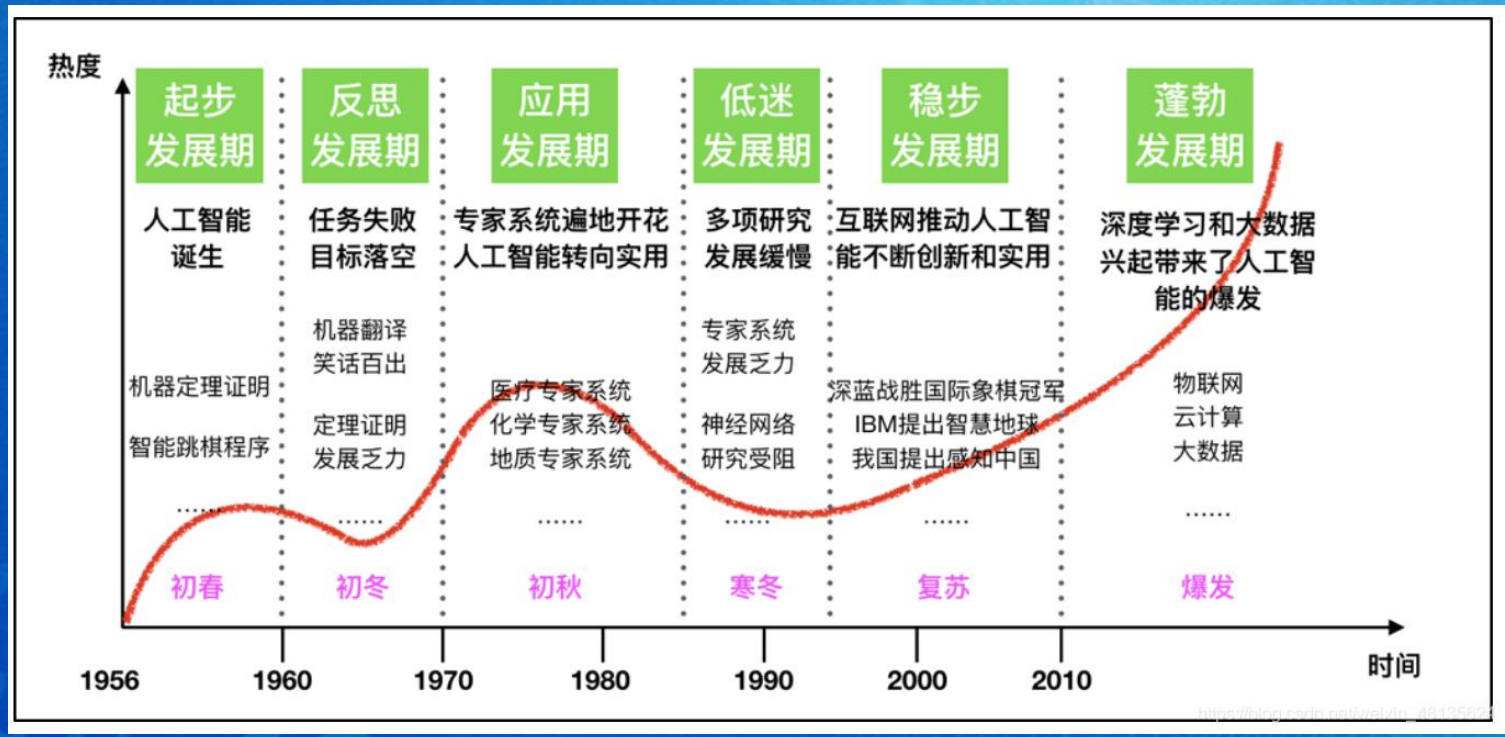
# 人工智能时代操作系统的机遇与挑战

余 杰 研究员

# 引言

## 人工智能发展历史

- 三次浪潮
- 人工智能进入大模型时代




2018-至今：大规模预训练模型

1950s-1990s：符号学习　1990s-2000s：统计学习　2010-2017：深度学习

# 引言

## 操作系统的三大任务

- 高效管理计算机的硬软件资源
- 为上层软件提供共性的基础服务
- 为用户提供友好易用的交互界面

**操作系统是计算机系统中的核心软件，是其他一切软件的基础**

## 三个发展阶段

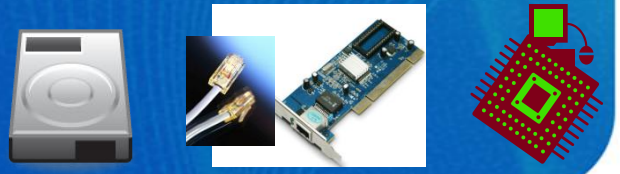
- 硬件的配套
  - 大型机时代，小型机时代
- 独立的软件产品
  - PC时代
  - Copyright Vs Copyleft
  - 闭源Vs开源
- 软件变服务
  - 控制生态系统的抓手

应用程序

向上提供共性服务
和人机交互界面


操作系统

向下管理资源



# 报告提纲

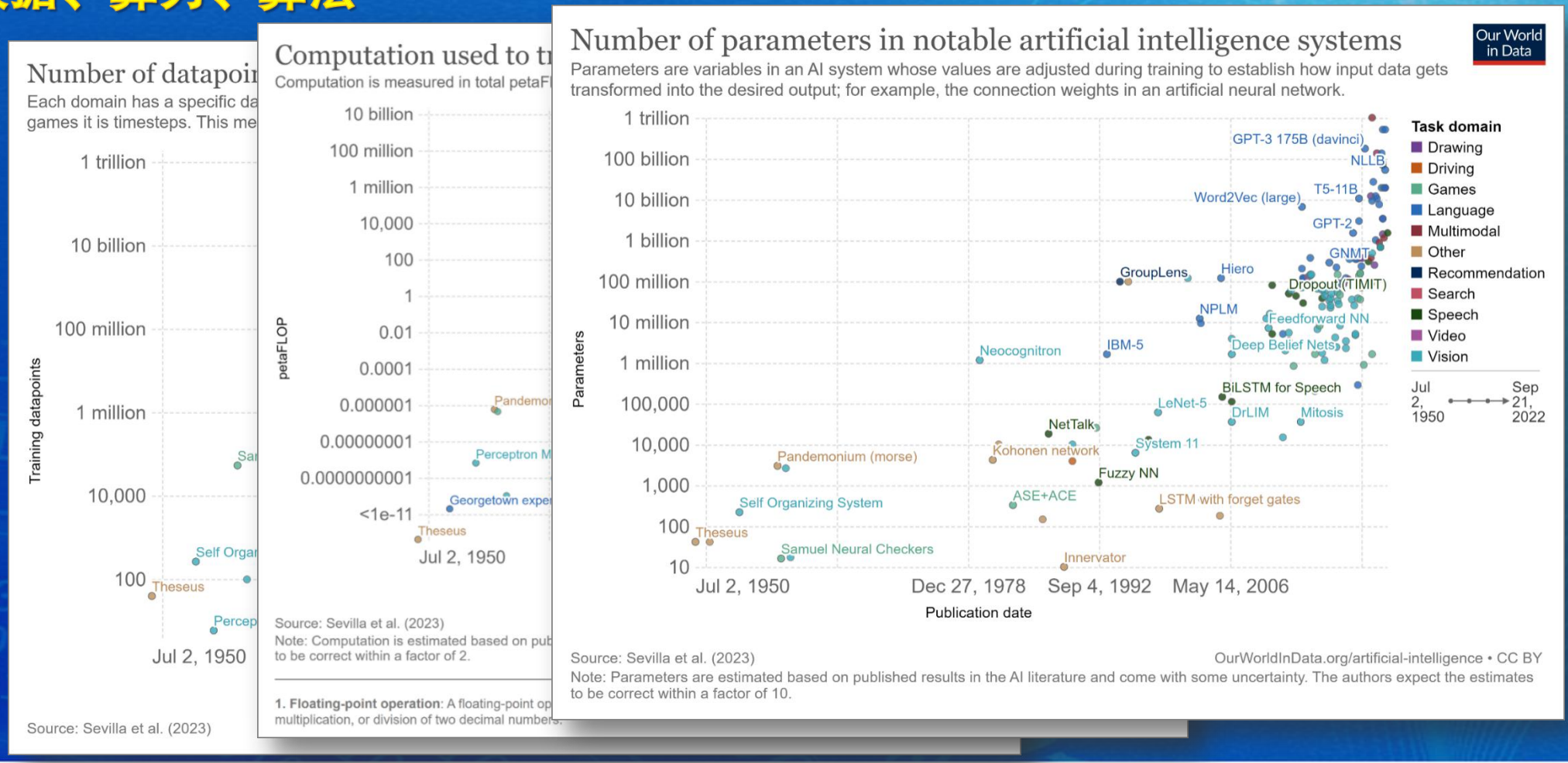
一、OS for AI:操作系统助力人工智能发展

二、OS with AI:操作系统智能化支撑技术

三、OS plus AI:云端一体化融合趋势

# 人工智能三要素

## 数据、算力、算法



# 人工智能三要素

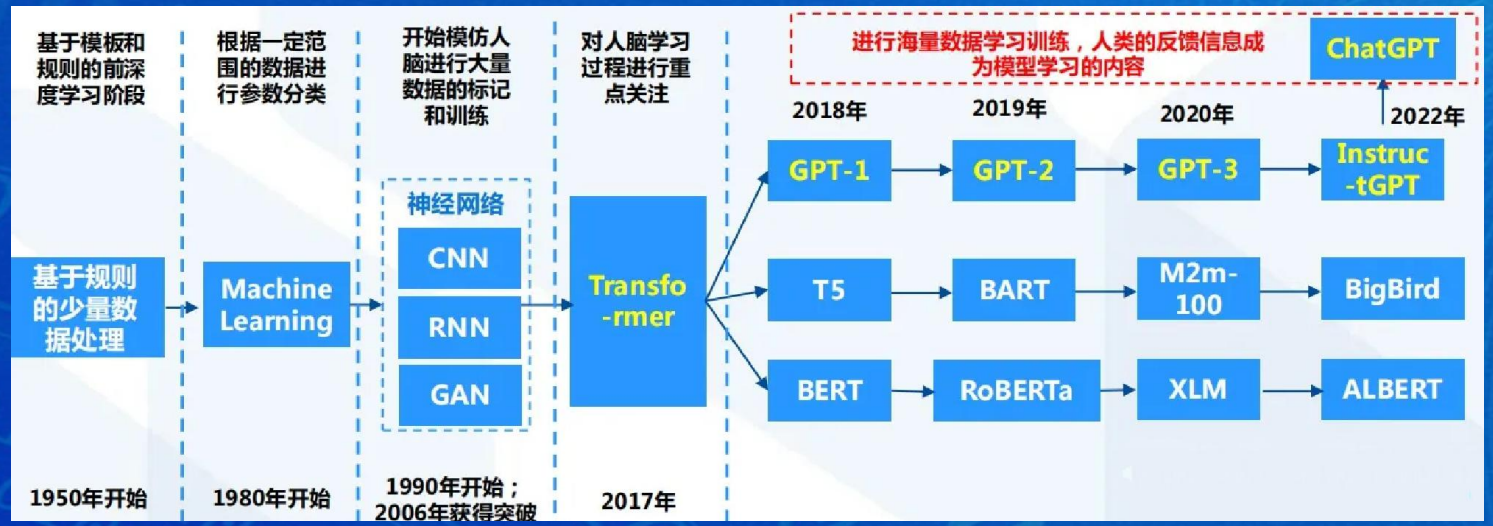
## 传统模型

- 传统模型：以专家系统、知识库为代表，具有一定知识推理能力，但规模较小，少量的专有知识（小数据）能在小算力的情况下通过集中式算法的处理，因此能够解决的问题也相对有限

## 大模型=大数据+大算力+强算法

- 大模型：在海量的泛化数据中，在大算力的支持下通过分布式算法驱动，使其具有泛化知识的推理模型

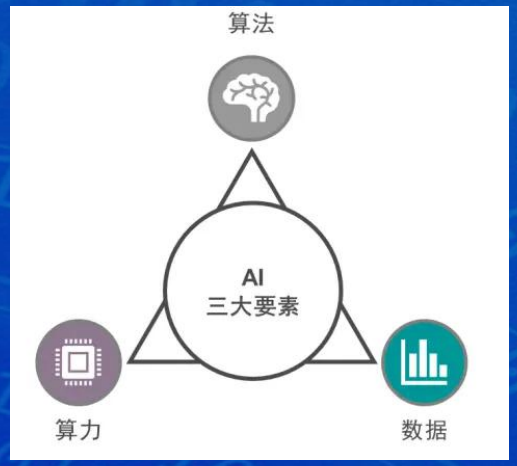
**ChatGPT**
技术路线



# 人工智能三要素

## 人工智能三要素：大数据，大算力和大模型

- 大数据：数据存储、数据管理、数据流通
- 大算力：硬件层（GPU、TPU），内核层（调度、驱动），基础库（glibc、gcc、python等）
- 大模型：大模型算法的提出和优化，一方面提升数据的利用率，同时也缓解算力的需求瓶颈




高质量数据

高效用算力

高效能算法

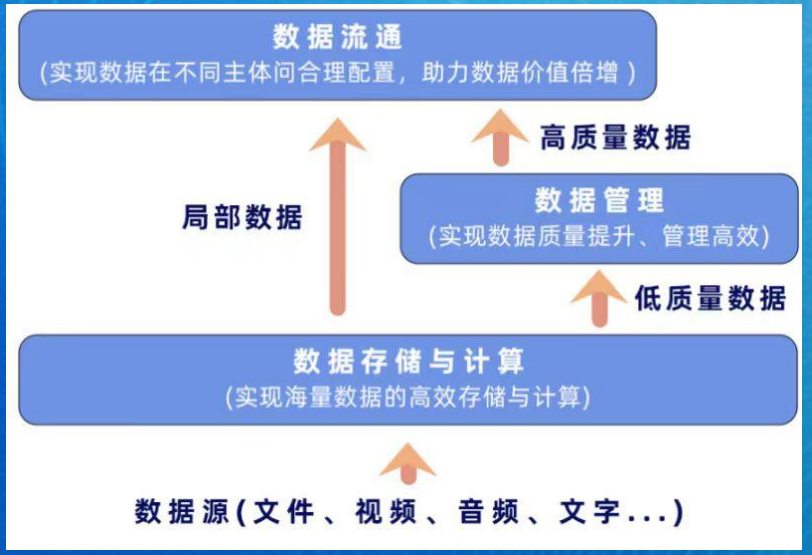
# OS for Data

## 大数据

- 传统的小样本数据难以满足大模型的需求，因此需要充足的大数据来训练和优化模型，大数据的广度和深度可以帮助模型更好地理解复杂的现象和问题，并提升其泛化能力

**大数据：实现高效学习和预测能力的基础**

- **数据存储与计算：数据的高效存储与计算，确保数据的可靠性和可用性**
- **数据管理：数据的有效管理，包括组织、维护和检索数据**
- **数据流通：按照需求在系统内部和外部的不同主体间流动**




操作系统在数据处理方面的作用是确保数据的高效存储、管理、访问、安全性和性能优化，同时支持多任务处理，以满足不同数据处理需求

# OS for Data

## 操作系统+大数据

- 专为大数据环境设计的**大数据操作系统**，致力于优化数据管理、加速分析处理，为用户提供更高效、可靠的数据操作体验

### 分布式计算与跨平台协同

- 旨在能够实现分布式计算的高效协同，同时能够无缝地与不同平台进行交互和整合
- 提供支持异构计算平台的操作系统架构，使得大数据处理能够更好地利用不同类型的硬件资源，提升计算效率


### 智能化数据管理与调度

- 通过整合先进的智能算法，在操作系统级别实现智能数据管理和任务调度，实现更高效的数据处理和分析
- 通过引入机器学习和智能决策，实现自适应性数据管理，使系统更好地适应不断变化的数据负载

[1] Himeur Y, Elnour M, Fadli F, et al. AI-big data analytics for building automation and management systems: a survey, actual challenges and future perspectives[J]. Artificial Intelligence Review, 2023, 56(6): 4929-5021.

# OS for Computing

## 大算力

- 大模型通常包含数百亿或数千亿的参数，需要大量的算力资源来训练和推理，强大的算力（例如高性能GPU、TPU和分布式计算集群）使大模型能够有效地进行复杂的运算，实现更准确和高效的任务执行

**大算力包括多种硬件和资源**

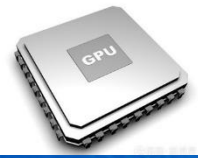
- **硬件层：**CPU、GPU、TPU、内存、网络、磁盘等硬件资源是大算力的基础
- **内核层：**驱动程序在操作系统内核层中起到关键作用，确保硬件和软件之间的协同工作
- **基础库：**glibc、gcc、python等基础库提供了支持大算力的编程工具和环境


龙芯

申威

英特尔、AMD、兆芯、海光

飞腾、鲲鹏




英伟达

操作系统对于大算力的核心作用在于：协调和优化计算资源的分配以提高计算性能，同时提供稳定的运行环境，以确保大算力系统的可靠性和高效性

# OS for Computing

## 操作系统+大算力

- 在追求巨大计算能力的背景下，操作系统与大算力的紧密结合打造了一种新的操作系统设计理念，专为高性能计算而设计，旨在提供无缝的、高度优化的计算环境

### 全栈化算力资源优化与智能调度

- **分布式大规模计算负载均衡：**引入高度智能化的负载均衡机制，自动检测和调整计算节点，确保各个节点间负载均衡，最大化大算力的利用效率
- **异构计算资源协同优化：**实现对异构计算资源的深度协同优化，智能地管理不同体系结构的处理器、加速器等，使其协同工作，以获得更大规模和更强大的计算能力
- **智能任务调度与预测性分析：**引入基于机器学习的任务调度和预测性分析，学习任务的运行特征，实现对任务完成时间的智能预测和最优调度，从而更有效地利用大算力

**可负担的算力成本**

算力成本高昂，需要充分利用异构算力，应对大规模算力需求

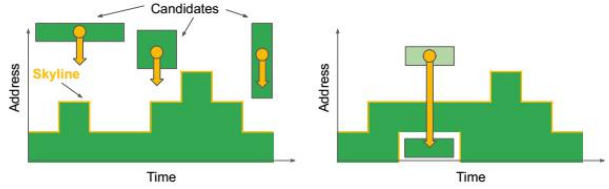
**解决方案：通过在操作系统中提供内核软件定义算力功能，以充分调度异构算力**

[1] Min C, Kang W, Kumar M, et al. Solros: a data-centric operating system architecture for heterogeneous computing[C]//Proceedings of the Thirteenth EuroSys Conference. 2018: 1-15.

# OS for Computing

## 优化内存分配来提供计算能力

- 传统的片上内存分配方案在处理机器学习任务时存在一些挑战，例如内存碎片化、内存冲突和动态内存需求，如何设计一个高效的**内存分配方案**成为一个具有挑战性的问题




(a) Simple placement heuristic. (b) Solver-guided placement.

Figure 8: TelaMalloc Placement Strategies.

### TelaMalloc 内存分配器

- TelaMalloc采用了一种基于矩形分割的空间划分策略，将片上内存划分为多个矩形块，并通过动态地分配和释放这些矩形块来满足不同任务的内存需求
- 在内存空间中维护一个分配表来跟踪内存块的分配情况，有效地管理内存资源，以减少内存碎片并提高内存利用率
- 使用一种动态内存调整算法，根据任务的实际内存需求来分配和释放矩形块，以解决内存冲突问题

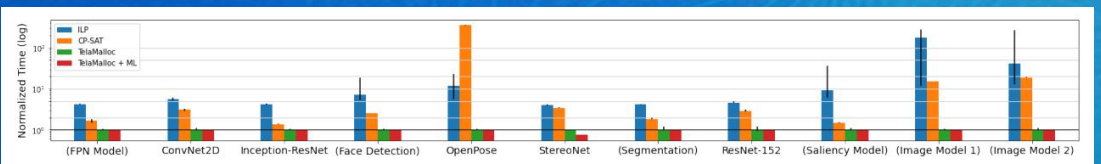



Figure 13: The same workloads running on a workstation, with additional CP-SAT and ML comparisons.

实验结果证明TelaMalloc能够更好地适应不同任务的动态内存需求，并有效地处理多个任务之间的内存冲突

[1] Maas M, Beaugnon U, Chauhan A, Ilbeyi B. TelaMalloc: Efficient On-Chip Memory Allocation for Production Machine Learning Accelerators. In Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1 2022 Dec 19 (pp. 123-137).

# OS for Model Dev

## 大模型开发

- 操作系统在大模型开发中发挥着至关重要的角色，提供**资源管理**、**性能监控**和**可扩展性**等关键功能，确保大模型的稳定训练和推理，实现高效、稳定和成本效益的大模型开发

**相较于传统机器学习开发，大模型开发所面临的挑战**

- **计算需求:** 大模型需要庞大的计算能力来训练和推理，通常依赖于GPU和其他专用硬件
- **数据需求:** 大模型需要海量数据来获得高质量的预测和决策，这增加了数据采集和处理的挑战
- **资源管理:** 有效管理硬件资源对于大模型至关重要，以确保计算机集群的充分利用
- **效率:** 为了降低训练时间和成本，大模型需要高效的算法和工作流程

**面向大模型开发的AI操作系统**

**（大模型和应用之间的桥梁）**

# OS for Model Dev

## 操作系统+大模型开发

- 操作系统与大规模模型开发的融合，开辟了一个全新的领域，致力于为大规模深度学习和模型训练提供最优化的支持，从而推动着先进模型的创新与发展

### 优化大规模模型开发

- **分布式模型训练与优化：**实现对大规模深度学习模型的分布式训练，有效协同多个计算节点，缩短训练时间，提升训练效率
- **模型部署与推理优化：**自动选择最优的硬件资源，并优化模型的推理路径，提高大规模模型在生产环境下的响应速度


- **智能化资源管理与调度：**根据模型需求和实际负载智能调整计算资源，提供高度灵活的资源管理，确保大规模模型开发的高效进行

[1] Torres-Sánchez E, Alastruey-Benedé J, Torres-Moreno E. Developing an AI IoT application with open software on a RISC-V SoC[C]//2020 XXXV Conference on Design of Circuits and Integrated Systems (DCIS). IEEE, 2020: 1-6.

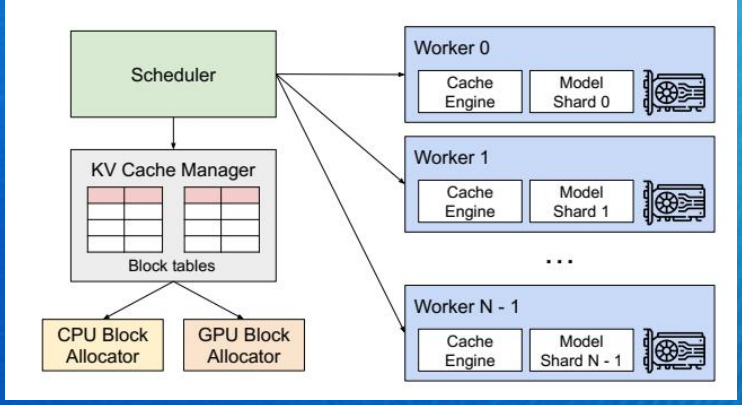
# OS for Model Dev

## 操作系统+大模型内存管理

- 大语言模型通常需要大量的内存来存储模型参数、输入和输出数据等，然而，在实际应用中，由于硬件资源有限，往往无法将所有数据一次性加载到内存中，导致推理速度变慢或者出现内存溢出等问题

### 基于分页的注意力机制PagedAttention

- PagedAttention技术将输入序列划分为多个页面，每个页面包含一定数量的词语或者子句
- 在计算注意力分数时，只计算当前页面内的词语与模型之间的交互，而忽略其他页面中的词语，这样就可以将大语言模型的内存需求降低到可接受的范围内




基于PagedAttention 的分页方式，缓存管理器通过集中调度器发送的指令来管理GPU Worker上的物理缓存

[1] Woosuk Kwon, et al. Efficient Memory Management for Large Language Model Serving with PagedAttention. SOSP2023.

# OS for Model Dev

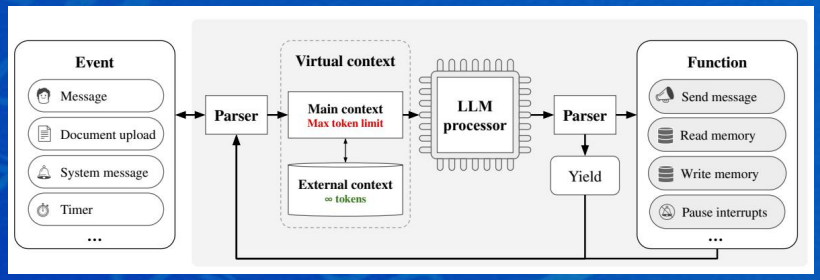
## 操作系统+大模型上下文窗口

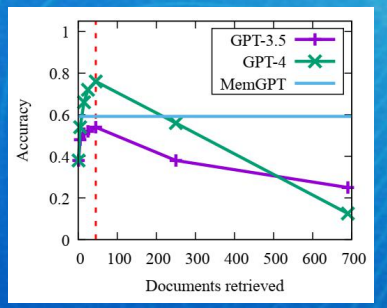
- 大模型为自然语言处理带来了巨大突破，然而，这些模型受限于固定的上下文窗口大小，在处理超出此范围的长文本和连续对话时，大模型在长篇文档分析和连续对话等方面的表现受到了限制，难以处理超出固定上下文范围的信息

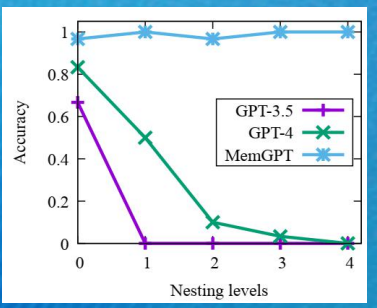
### 虚拟上下文管理技术

- 传统OS中，层次化内存通过将数据在快速和慢速内存之间移动，为系统提供了扩展的内存容量
- 参考OS中的内存管理，提出虚拟上下文管理技术，扩展大模型的上下文范围，不再受限于固定上下文窗口

**Memory-GPT系统**：通过智能管理不同内存层级，能够在有限上下文窗口内获取扩展的上下文信息，从而提高其在长文本分析和连续对话等任务中的性能



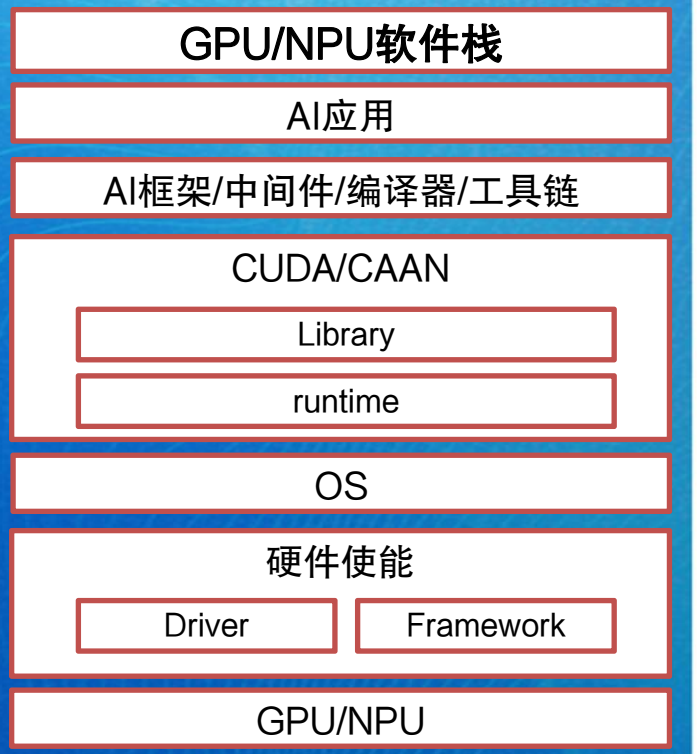

实验结果显示，在文档问答（Document QA）和嵌套键值检索（nested KV retrieval）两种任务中，相比GPT-3.5和GPT-4，Memory-GPT的准确性能不受上下文长度增加的影响

[1] Packer, Charles, et al. "MemGPT: Towards LLMs as Operating Systems." arXiv preprint arXiv:2310.08560 (2023).

# OS for AI

## 开源操作系统支撑多种硬件使能和通用计算框架

- Red Hat
  - 支持开源机器学习和深度学习库，如TensorFlow、PyTorch、Scikit-Learn等
  - 提供了Red Hat OpenShift AI，用于为人工智能应用构建、训练、测试和提供模型
  - Ansible LightSpeed ，将生成式AI集成进Ansible，用于运维自动化操作和部署
- Debian/Ubuntu
  - 提供了相应的驱动程序支持，保证与CUDA和NVIDIA GPU的完全兼容性
  - 支持流行的机器学习框架、库和工具，如TensorFlow、PyTorch、Scikit-Learn 等
  - 提供容器化的机器学习解决方案，增加可扩展性和部署效率
  - 为IoT（物联网）和边缘设备提供了专门的版本，称为Ubuntu Core



# 报告提纲

一、OS for AI:操作系统助力人工智能发展

二、OS with AI:操作系统智能化支撑技术

三、OS plus AI:云端一体化融合趋势

# 学术界思想活跃，以AI技术提升系统处理能力

## AI技术+内核决策（ASPLOS 23）

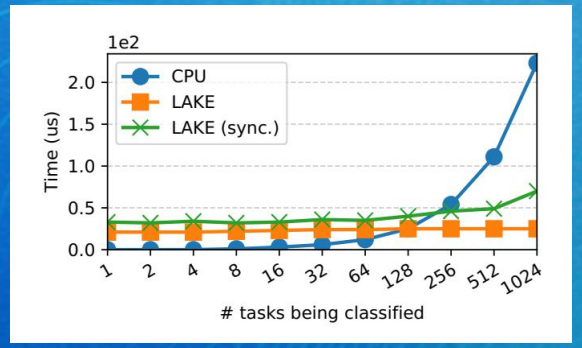
- 传统的内核决策往往依赖于手工调优的启发式算法，需要针对特定场景进行硬编码的决策规则，**机器学习**技术的引入可以改进内核决策和管理的方式

### 在内核空间中使用机器学习技术

- 会存在专用硬件的可访问性、数据收集和特征提取、资源竞争与冲突和可迁移性和适应性等挑战
- 需要设计和实现适用于内核空间的机器学习框架和接口，使得机器学习算法能够有效地集成到内核中

### LAKE: 支持机器学习辅助的内核系统

1. 提供一套API和机制，用于在内核空间中收集、管理和提取适合机器学习的特征
2. 使用机器学习算法来替代传统的启发式决策方法
3. 解决内核空间中的资源竞争和冲突问题
4. 支持使用专用硬件如GPU来加速机器学习决策过程
5. 可以适应不同的硬件平台和操作系统环境




以负载均衡实验为例，机器学习辅助的内核决策能够显著提高系统性能、资源利用效率和用户体验

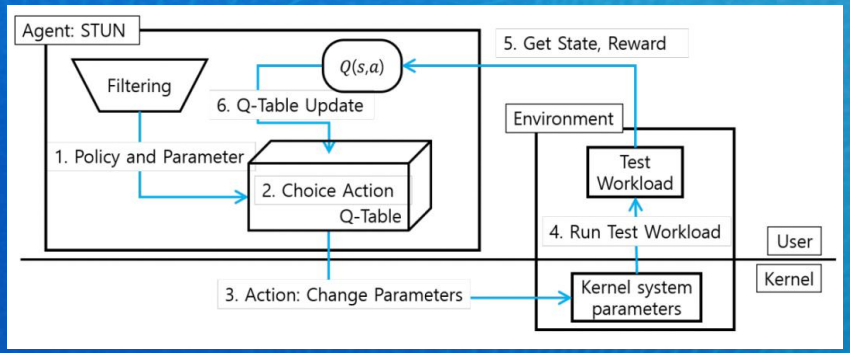
[1] Fingler H, Tarte I, Yu H, Szekely A, Hu B, Akella A, Rossbach CJ. Towards a Machine Learning-Assisted Kernel with LAKE. ASPLOS2023.

# 学术界思想活跃，以AI技术提升系统处理能力

## AI技术+进程调度（Applied Sciences 2022）

- **调度参数：**Linux内核中提供了14个调度器参数进行优化，其中5个参数不影响性能，其他9个参数可以更改的参数值范围和Linux内核的默认值
- **传统方法：**传统进程调度算法通常基于固定的优先级或时间片轮转策略
- **AI技术：**如何使用机器学习算法来学习任务特征和工作负载的变化，并根据这些信息进行动态调整和优化进程调度策略

| Parameter | Default Value | Range |
|---|---|---|
| latency_ns | 24,000,000 | 100,000~1,000,000,000 (1 s) |
| migration_cost_ns | 500,000 | 0~1,000,000,000 |
| min_granularity_ns | 3,000,000 | 100,000~1,000,000,000 (1 s) |
| nr_migrate | 32 | 0~ |
| rr_timeslice_ms | 100 | 0~1000 |
| rt_period_us | 1,000,000 | 900,000~1,000,000 |
| rt_runtime_us | 950,000 | 0~1,000,000 |
| cfs_bandwidth_slice_us | 5000 | 0~1,000,000 |
| wakeup_granularity_ns | 4,000,000 | 0~1,000,000,000 |




### 基于强化学习的内核进程调度策略

- **状态：**包括当前任务队列、CPU和GPU利用率等信息
- **动作：**动作空间定义为可供选择的进程调度动作
- **奖励函数：**奖励函数设计为使得响应时间最小化且资源利用率最大化的联合目标函数

[1] Lee H, Jung S, Jo H. STUN: Reinforcement-Learning-Based Optimization of Kernel Scheduler Parameters for Static Workload Performance. Applied Sciences. 2022; 12(14):7072.

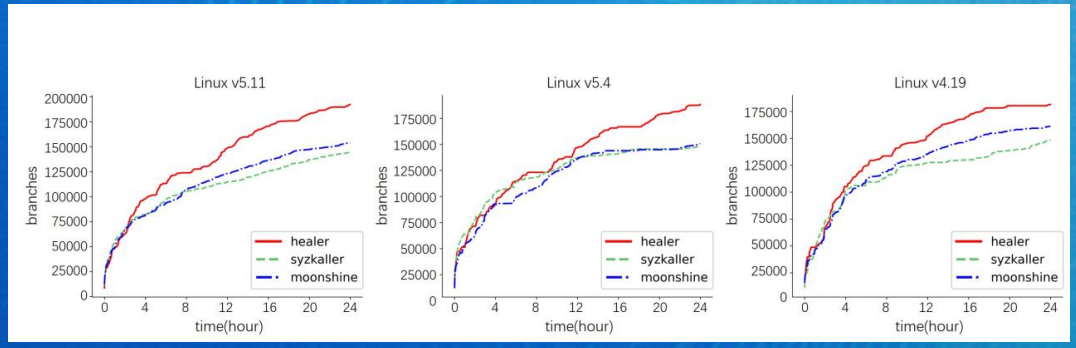
# 学术界思想活跃，以AI技术提升系统处理能力

## AI技术+内核模糊测试（SOSP 21）

- 内核模糊测试是一种发现内核漏洞和安全问题的方法，传统模糊测试存在很大的局限性：一是测试样本的数量巨大，测试效率低下；二是由于内核的复杂性，很难准确地生成有效的测试用例

### HEALER：利用关系学习来引导内核模糊测试

- 旨在通过学习数据中的关系和模式，进行预测和推理，在内核模糊测试中应用关系学习来生成有效的输入
- HEALER的三个关键组件：
  - **基于图的表示**：将内核操作转换为图结构，以捕获操作之间的关系
  - **关系学习模型**：使用机器学习技术学习内核操作之间的关系
  - **引导模糊测试**：根据关系学习模型指导模糊测试生成输入




实验结果表明，与目前主流的内核模糊测试相比，Healer平均提升28%与21%的分支覆盖率，并且效率提升了2.2倍与1.8倍

[1] Hao Sun, Yuheng Shen, Cong Wang, Jianzhong Liu, Yu Jiang, Ting Chen, and Aiguo Cui. HEALER: Relation Learning Guided Kernel Fuzzing. SOSP2021.

# 学术界思想活跃，以AI技术提升系统处理能力

## AI技术+内核数据路径（HotOS 21）

- 在传统的操作系统中，内核功能通常由固定的数据路径实现，难以适应不同的应用需求和硬件平台，且随着计算任务的复杂化和数据量的增加，传统的静态优化方法已经无法充分发挥硬件的潜力

**基于学习优化的可重构内核数据路径设计方法**

- 利用机器学习和深度学习算法，对大量数据进行训练和学习，以自动发现和应用最佳的优化策略，实现性能提高、功耗降低和加速计算过程
  - 使用深度强化学习技术，对内核操作进行建模和训练，使得内核数据路径能够在运行时自动调整和优化
  - 操作系统根据实际需求和硬件条件，动态选择最优的数据路径来执行内核操作，从而提高系统的性能和效率

| Metric \ Benchmark | OpenCV video resize | | | Numpy matrix conv | | |
|---|---|---|---|---|---|---|
| | Linux | Leap | Ours | Linux | Leap | Ours |
| Accuracy (%) | 40.69 | 45.40 | **78.89** | 12.50 | 48.86 | **92.91** |
| Coverage (%) | 65.09 | 66.81 | **84.13** | 19.28 | 65.62 | **88.51** |
| Completion time (s) | 24.60 | 23.02 | **17.79** | 31.74 | 17.48 | **13.90** |

**Table 1: Case study: Page prefetching.**

实验结果显示：机器学习模型提高了准确率且显著缩短了作业完成时间

[1] Qiu Y, Liu H, Anderson T, Lin Y, Chen A. Toward reconfigurable kernel datapaths with learned optimizations. InProceedings of the Workshop on Hot Topics in Operating Systems 2021 Jun 1 (pp. 175-182).

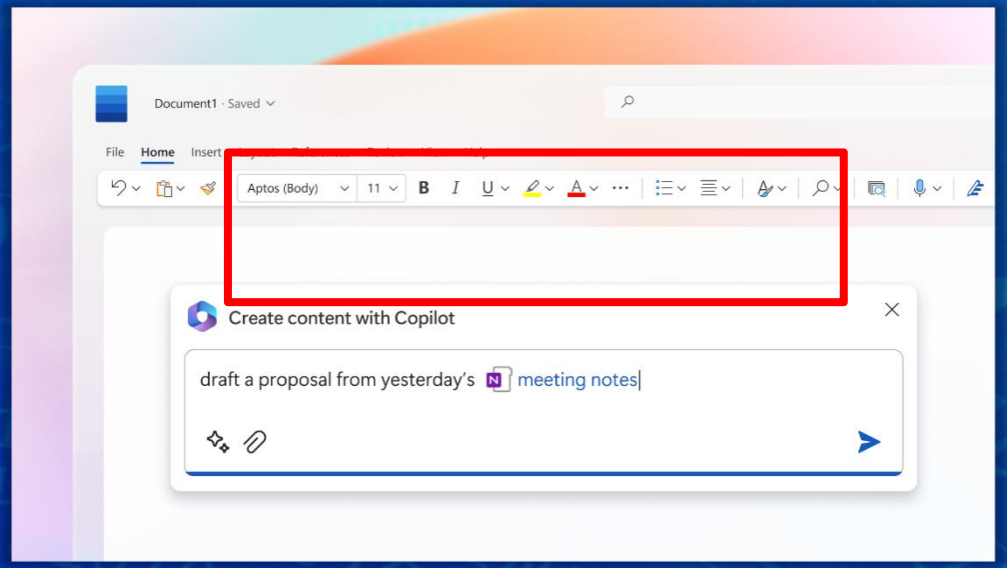
# 主流OS厂商拥抱大模型，探索人机交互的新模式

**微软**

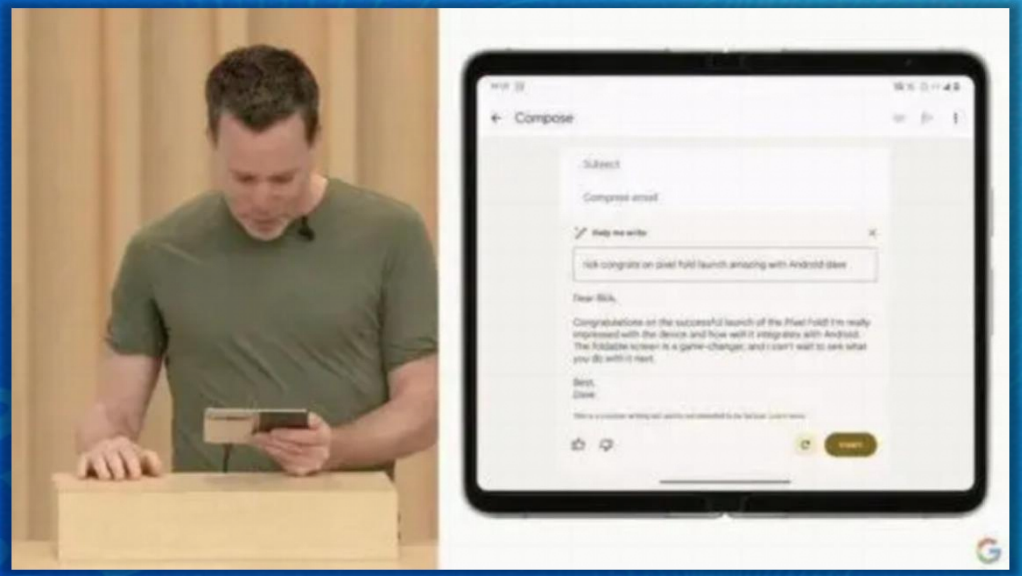
- 2023年5月，微软宣布在Windows 11中加入AI助手Copilot，9月，微软将Copilot的强大能力嵌入到Word和Excel等核心生产力应用程序当中

**谷歌**

- 2023年5月，谷歌最新的Android 14中，集成许多AI功能，包括Magic Compose (魔法撰写)和Cinematic Wallpapers（电影壁纸）和Generative AI Wallpapers （生成式AI壁纸）




微软AI助手Copilot

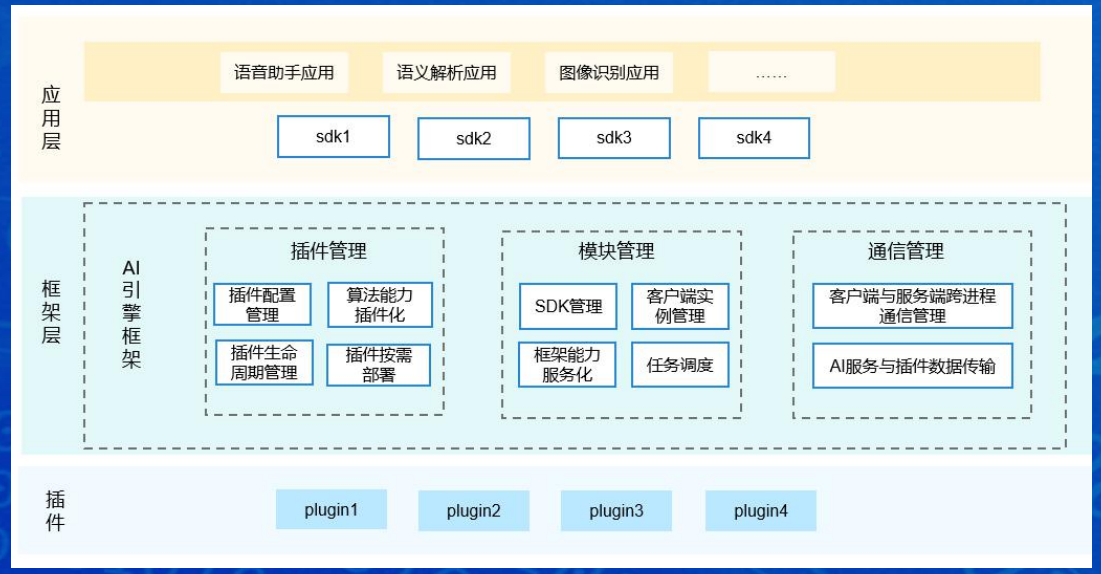



Android 14魔法撰写

# 主流OS厂商拥抱大模型，探索人机交互的新模式

## 华为（小米、vivo、oppo类似）

- 2021年3月，OpenHarmony 新增AI业务子系统，作为提供原生分布式AI能力的子系统，开源统一的AI引擎框架，可实现算法能力快速插件化集成




OpenHarmony AI引擎框架

## 麒麟

- 2023年7月，openKylin 1.0版本中支持桌面AI大模型插件，实现基于大语言模型的聊天机器人功能。此外，也支持智能语音助手功能，用户通过语音指令即可触发应用功能
- 2023年11月，openKylin发布AI框架安装助手，具备智能推荐、一键自动、无需值守、过程可见、节省资源等特点


openKylin AI助手、AI框架安装助手

# 报告提纲

一、OS for AI:操作系统助力人工智能发展

二、OS with AI:操作系统智能化支撑技术

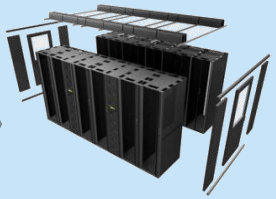
三、OS plus AI:云端一体化融合趋势

# 三、OS plus AI:云端一体化融合趋势

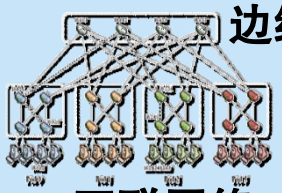
- 操作系统发展趋势：云端一体化
  - 人机物融合催生云边端协同的操作系统发展需求

资源虚拟化
分布式文件系统
……

云、边



基础架构



互联网络

边缘设备

融合业务系统


网络优化等

网络


有线网络技术
10G/40G/100G以太网

互联网
移动网络
企业专网
……

无线网络技术
WiFi 11n / 3G LTE

新型人机交互等

端

呼叫中心

会议系统

协同终端

办公终端

3G终端

传感器

# 三、OS plus AI:云端一体化融合趋势

IT基础设施云端一体化融合

## 主要研究领域

- 云端一体化：支撑分布式AI
- 云端一体化：加速AI推理
- 云端一体化：助力AI开发

KDD ICLR

## 主流芯片厂商

- 推动大模型的小型化
- 端侧AI
- 混合AI

Qualcomm 高通 Lenovo

OS plus AI

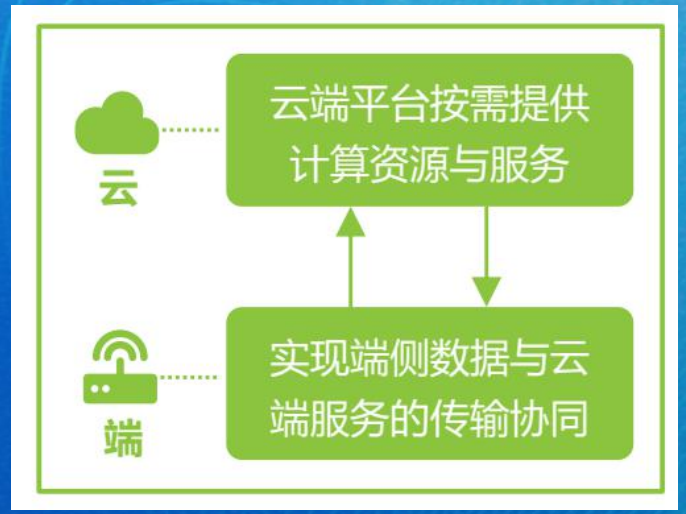
# 云端一体化：支撑分布式AI

## 云端一体化+分布式AI

- 融合云、边缘计算与分布式人工智能，云边端一体化不仅实现了数据和计算资源的高效整合，更为分布式AI注入了灵活性和强大的计算能力，推动着智能化应用的发展

### 云端一体化在支持分布式AI方面的优势

- **高效的计算资源管理：**将云计算和本地资源融合，实现对计算资源的智能化管理和分配，使分布式AI能够更有效地利用多个计算节点，加速任务的执行
- **大规模数据处理：**打破数据和通信的隔阂，实现数据的即时传输和分布式计算，为分布式AI提供了更灵活的环境，使其能够更快速地响应不同数据源的变化
- **强大的规模化应用能力：**支持分布式AI的规模化部署，使得在大规模数据集上进行训练和推理成为可能，这种规模化应用能力为复杂的深度学习任务提供了可行性




[1] Duan, Sijing, Dan Wang, Ju Ren, Feng Lyu, Ye Zhang, Huaqing Wu, and Xuemin Shen. "Distributed artificial intelligence empowered by end-edge-cloud computing: A survey." IEEE Communications Surveys & Tutorials (2022).

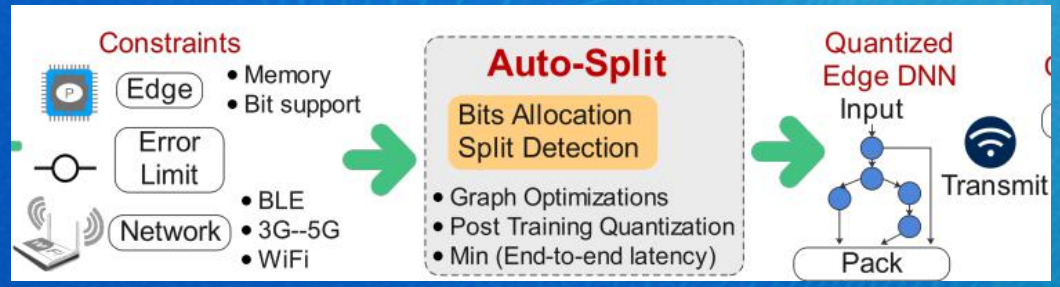
# 云端一体化：加速AI推理

## 云端一体化+AI框架

- 一方面，为了提高AI推理的效率和响应速度，研究人员提出一种AI通用框架Auto-Split，通过将AI任务自动分割成边缘设备和云服务器之间的最小可执行单元，从而满足AI推理的加速需求

### Auto-Split推理框架

- 首先，对于一个输入样本，Auto-Split首先将其分割成多个子样本，并在边缘设备上执行预处理和特征提取
- 然后，将这些子样本发送到云服务器进行后处理和模型推理
- 最后，合并云服务器的推理结果，形成最终的输出




[1] Banitalebi-Dehkordi, Amin, et al. "Auto-split: A general framework of collaborative edge-cloud AI." Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021.

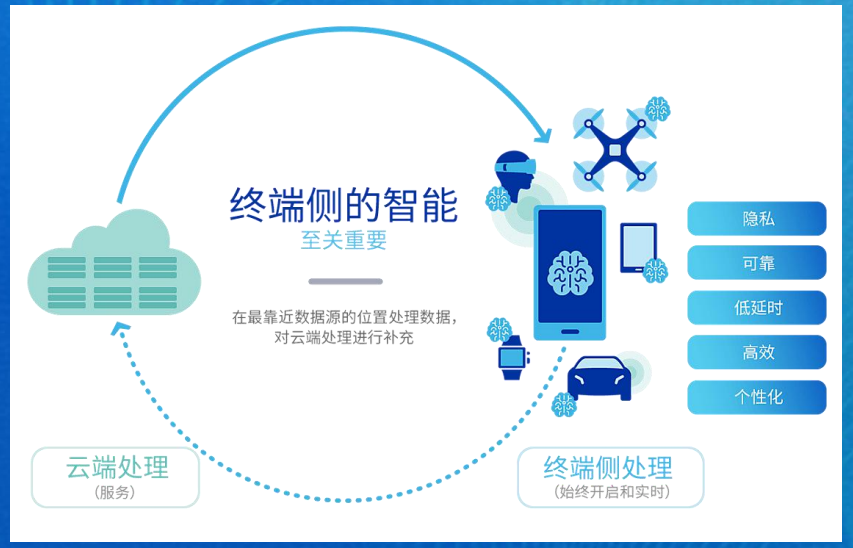
# 云端一体化：加速AI推理

## 云端一体化+端侧AI

- 另一方面，完全依靠云端部署和应用AI成本相对高昂，在云端一体化融合下，可将部分AI处理从云端卸载至端侧计算，在减轻云端网络负担、高可靠性和低时延场景以及隐私和安全方面具有应用必要性，有助于推动大模型的全面应用

### 云端一体化融合加速端侧AI

- **减轻网络负担：**利用已经部署的手机、PC 等终端设备，开展端侧AI计算，成本上将大幅节约，也有助于降低云厂商的能源消耗
- **高可靠性与低时延优势：**云端连接在访问拥挤时将产生高延迟，甚至会被拒绝服务，通过将计算负载迁移到端侧进行，可靠性和低时延优势明显
- **隐私和安全保障：**不同于访问云端的数据交互，端侧AI中所计算的数据具有更强的私密性和安全保障




[1] Liu, Deyin, et al. "HierTrain: Fast hierarchical edge AI learning with hybrid parallelism in mobile-edge-cloud computing." IEEE Open Journal of the Communications Society 1 (2020): 634-645.

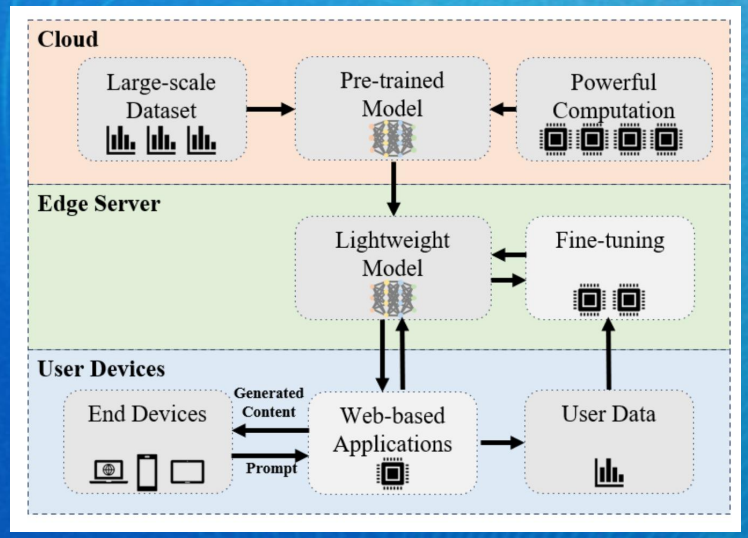
# 云端一体化：助力AI开发

## 云端一体化+生成式AI

- 云端一体化融合将生成式AI与云计算有效结合，提供强大的计算能力和大规模数据处理，从而加速生成式AI的训练和应用，这种融合不仅加强了性能，还使生成式AI更具可扩展性，能够更广泛地服务于各个领域

### 云端一体化提升生成式AI性能

- **增强计算能力：** 云端一体化使得生成式AI能够利用云计算集群的强大计算资源，加速模型训练和推理
- **大规模数据处理：** 云端存储和处理大规模数据，使得生成式AI能够更好地学习和理解复杂的模式
- **实时应用支持：** 云端一体化允许生成式AI在实时环境中进行应用，如实时翻译、语音生成等，提升用户体验




基于云边端一体化的生成式AI

[1] Wang, Yun-Cheng, et al. "An Overview on Generative AI at Scale with Edge-Cloud Computing." (2023).

# 云端一体化：助力AI开发

## 大模型参数微调（基于重参数化的方法）

### AdaLoRA

**SVD形式参数更新**

$$W = W^{(0)} + \Delta = W^{(0)} + P\Lambda Q$$

$$R(P,Q) = \|P^\top P - I\|_F^2 + \|QQ^\top - I\|_F^2$$

基于重要程度的参数分配

| Method | # Params | MNLI m/mm | SST-2 Acc | CoLA Mcc | QQP Acc/F1 | QNLI Acc | RTE Acc | MRPC Acc | STS-B Corr | All Ave. |
|---|---|---|---|---|---|---|---|---|---|---|
| Full FT | 184M | 89.90/90.12 | 95.63 | 69.19 | **92.40/89.80** | 94.03 | 83.75 | 89.46 | 91.60 | 88.09 |
| BitFit | 0.1M | 89.37/89.91 | 94.84 | 66.96 | 88.41/84.95 | 92.24 | 78.70 | 87.75 | 91.35 | 86.02 |
| HAdapter | 1.22M | 90.13/90.17 | 95.53 | 68.64 | 91.91/89.27 | 94.11 | 84.48 | 89.95 | 91.48 | 88.12 |
| PAdapter | 1.18M | 90.33/90.39 | 95.61 | 68.77 | 92.04/89.40 | 94.29 | 85.20 | 89.46 | 91.54 | 88.24 |
| LoRA$_{r=8}$ | 1.33M | 90.65/90.69 | 94.95 | 69.82 | 91.99/89.38 | 93.87 | 85.20 | 89.95 | 91.60 | 88.34 |
| AdaLoRA | 1.27M | **90.76/90.79** | **96.10** | **71.45** | **92.23/89.74** | **94.55** | **88.09** | **90.69** | **91.84** | **89.31** |
| HAdapter | 0.61M | 90.12/90.23 | 95.30 | 67.87 | 91.65/88.95 | 93.76 | 85.56 | 89.22 | 91.30 | 87.93 |
| PAdapter | 0.60M | 90.15/90.28 | 95.53 | 69.48 | 91.62/88.86 | 93.98 | 84.12 | 89.22 | 91.52 | 88.04 |
| HAdapter | 0.31M | 90.10/90.02 | 95.41 | 67.65 | 91.54/88.81 | 93.52 | 83.39 | 89.25 | 91.31 | 87.60 |
| PAdapter | 0.30M | 89.89/90.06 | 94.72 | 69.06 | 91.40/88.62 | 93.87 | 84.48 | 89.71 | 91.38 | 87.90 |
| LoRA$_{r=2}$ | 0.33M | 90.30/90.38 | 94.95 | 68.71 | 91.61/88.91 | 94.03 | 85.56 | 89.71 | **91.68** | 88.15 |
| AdaLoRA | 0.32M | **90.66/90.70** | **95.80** | **70.04** | **91.78/89.16** | **94.49** | **87.36** | **90.44** | 91.63 | **88.86** |

### QLoRA

**4位NormalFloat量化、双重量化、分页优化器**

- 将65B模型的微调平均内存需求从>780GB的GPU内存降低到<48GB，不会降低运行时间或预测性能
- 加载QLORA训练得到的7B模型仅仅需要3G的显存，真正做到了在消费级显卡上都能跑大模型

| Model / Dataset | Params | Model bits | Memory | ChatGPT vs Sys | Sys vs ChatGPT | Mean | 95% CI |
|---|---|---|---|---|---|---|---|
| GPT-4 | - | - | - | 119.4% | 110.1% | **114.5%** | 2.6% |
| Bard | - | - | - | 93.2% | 96.4% | 94.8% | 4.1% |
| **Guanaco** | 65B | 4-bit | 41 GB | 96.7% | 101.9% | **99.3%** | 4.4% |
| Alpaca | 65B | 4-bit | 41 GB | 63.0% | 77.9% | 70.7% | 4.3% |
| FLAN v2 | 65B | 4-bit | 41 GB | 37.0% | 59.6% | 48.4% | 4.6% |
| **Guanaco** | 33B | 4-bit | 21 GB | 96.5% | 99.2% | **97.8%** | 4.4% |
| Open Assistant | 33B | 16-bit | 66 GB | 91.2% | 98.7% | 94.9% | 4.5% |
| Alpaca | 33B | 4-bit | 21 GB | 67.2% | 79.7% | 73.6% | 4.2% |
| FLAN v2 | 33B | 4-bit | 21 GB | 26.3% | 49.7% | 38.0% | 3.9% |
| Vicuna | 13B | 16-bit | 26 GB | 91.2% | 98.7% | **94.9%** | 4.5% |
| **Guanaco** | 13B | 4-bit | 10 GB | 87.3% | 93.4% | 90.4% | 5.2% |
| Alpaca | 13B | 4-bit | 10 GB | 63.8% | 76.7% | 69.4% | 4.2% |
| HH-RLHF | 13B | 4-bit | 10 GB | 55.5% | 69.1% | 62.5% | 4.7% |
| Unnatural Instr. | 13B | 4-bit | 10 GB | 50.6% | 69.8% | 60.5% | 4.2% |
| Chip2 | 13B | 4-bit | 10 GB | 49.2% | 69.3% | 59.5% | 4.7% |
| Longform | 13B | 4-bit | 10 GB | 44.9% | 62.0% | 53.6% | 5.2% |
| Self-Instruct | 13B | 4-bit | 10 GB | 38.0% | 60.5% | 49.1% | 4.6% |
| FLAN v2 | 13B | 4-bit | 10 GB | 32.4% | 61.2% | 47.0% | 3.6% |
| **Guanaco** | 7B | 4-bit | 5 GB | 84.1% | 89.8% | **87.0%** | 5.4% |
| Alpaca | 7B | 4-bit | 5 GB | 57.3% | 71.2% | 64.4% | 5.0% |
| FLAN v2 | 7B | 4-bit | 5 GB | 33.3% | 56.1% | 44.8% | 4.0% |

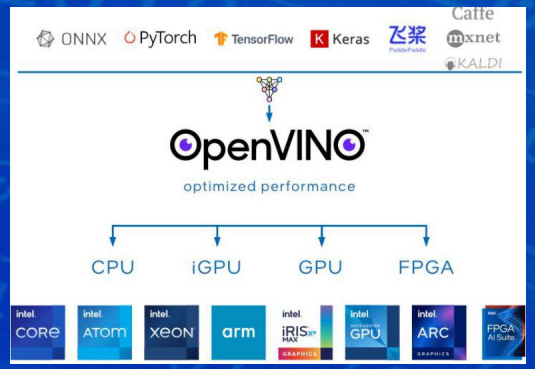
[1] Zhang, Qingru, et al. "Adaptive budget allocation for parameter-efficient fine-tuning." ICLR2023.
[2] Dettmers, Tim, et al. "Qlora: Efficient finetuning of quantized llms." arXiv preprint arXiv:2305.14314 (2023).

# 主流芯片厂商，推动大模型的小型化，混合AI

- **高通：混合AI**
  - **全栈式AI：**高通已经在终端AI相关硬件（高通AI引擎）、软件（高通AI软件栈）、生态（数十亿终端设备）等方面储备了诸多产品和技术，形成了自己的全栈式AI能力
  - **混合AI：**2023年6月发布白皮书《混合AI是AI的未来》，提出了混合AI架构的概念，未来生成式AI的发展必然会是云端与终端侧的“混合”模式
  - **生成式AI落地终端：**一部搭载高通第二代骁龙8移动平台的智能手机，已经可以直接在本地运行参数超过10亿的文本到图像生成式AI模型Stable Diffusion，并且在15秒内生成一张512x512像素的图像，其时延已经可以做到和云端相当，实现在智能手机本地运行大模型


# 主流芯片厂商，推动大模型的小型化，混合AI

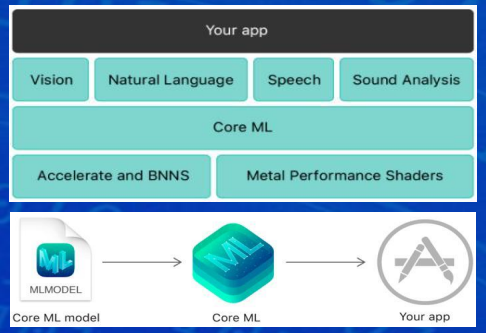
- Intel：AI PC
  - OpenVINO（Open Visual Inference and Neural network Optimization）开源工具套件，旨在优化深度学习模型的推理性能，提供了一种跨平台、高性能的部署解决方案，使得深度学习模型能够在各种硬件平台上高效运行，包括英特尔的CPU、GPU、FPGA、视觉处理器等
  - 2023年9月，Intel推出首款AI PC处理器酷睿Ultra，率先推动其AI PC加速计划，预计将在2025 年前为超过1 亿台终端PC带来AI特性
  - 具备离线情况下生成式AI的处理能力（首款搭载的笔记本宏碁Swift预计12月14日上市）
  - 希望将高能效AI技术与PC深度融合，进而从根本上改变、重塑PC体验





# 主流芯片厂商，推动大模型的小型化，混合AI

- 苹果
  - CoreML是嵌入iOS系统的机器学习框架
    - 利用CPU、GPU和神经引擎来优化设备上的性能，同时最大程度地减少其内存占用空间和功耗
    - 在用户设备上运行模型，将消除对网络连接的需求，并保持用户数据的私密性和应用程序的响应速度
    - CoreML支持视觉处理、自然语言、speech转换音频文本、音频识别的核心模型、以及用户自己训练各种模型
  - iOS 17进一步优化了设备的AI功能，包括更强大的机器学习和深度学习支持
    - 让用户可以更好地利用设备进行文本联想、语音实时转换、图像和视频处理
  - A17 Pro 芯片可更高效地支持机器学习算法，提升在设备上本地运行的生成式 AI 应用程序的能力



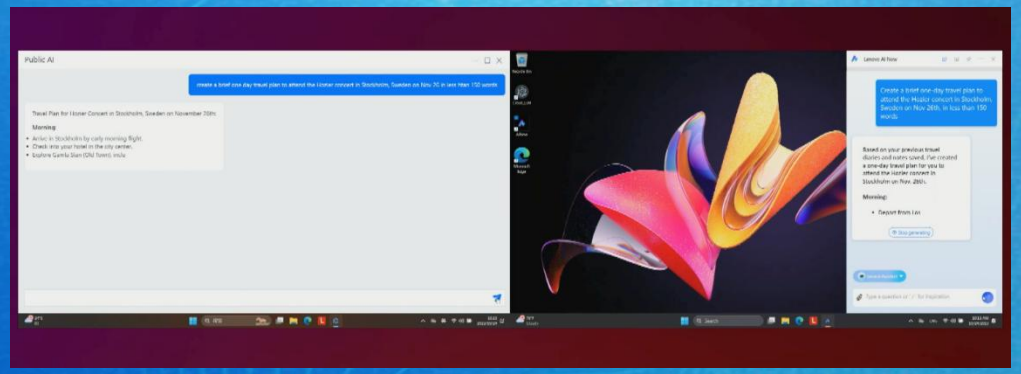

Core Machine Learning框架


iOS利用AI克隆自己的声音


苹果A17 Pro 芯片

# 主流芯片厂商，推动大模型的小型化，混合AI

- **联想**
  - AI PC：2023年10月，联想展示了首款AI PC产品，AI PC能够创建个性化的本地知识库，通过模型压缩技术运行个人大模型，实现AI自然交互



联想 PC 大模型与云端大模型并列演示

- **联发科**
  - **AI处理器APU**：2023年11月，发布天玑9300，搭载联发科第七代AI处理器APU 790,为生成式AI而设计,拥有硬件级的生成式AI引擎,可以实现更加高速且安全的边缘AI计算
  - **端侧生成式AI**：天玑9300支持端侧运行生成式AI，在vivo旗舰手机端侧落地70亿参数AI大语言模型，处理速度可达20 Tokens每秒


联发科第七代AI处理器APU 790

openKylin

# 敬请批评指正！

yj@nudt.edu.cn