

计算图 挑战和未来



ZOMI



关于本内容

1. 内容背景

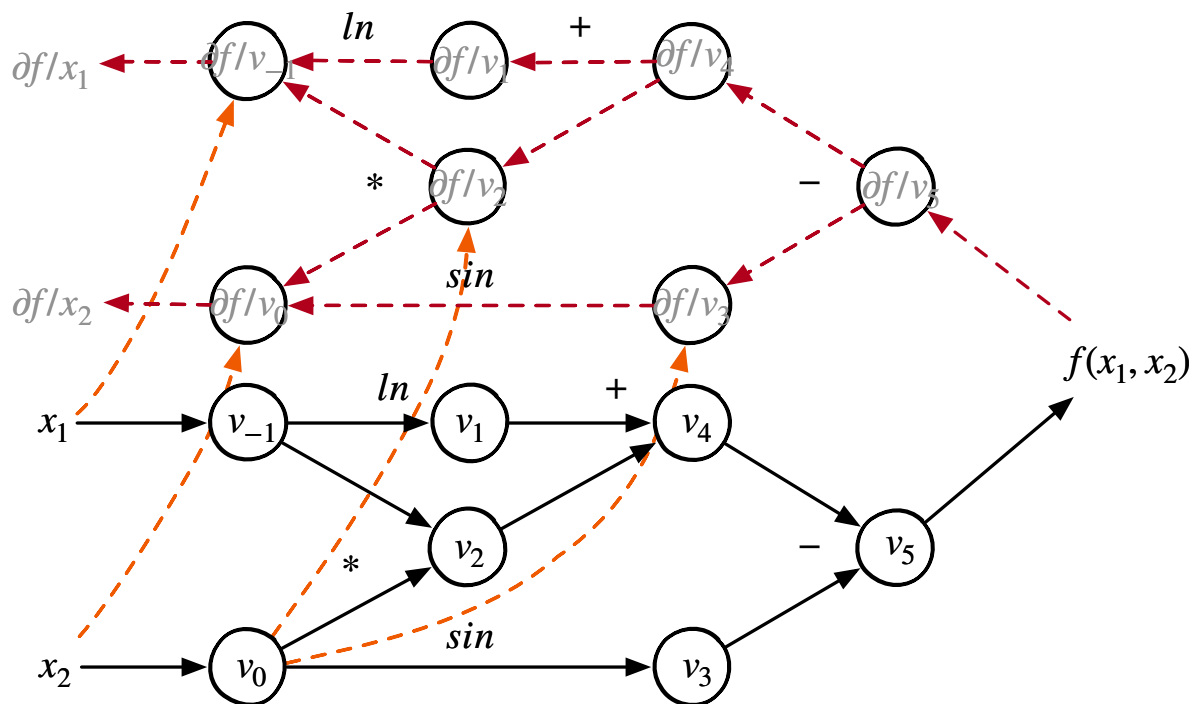
- 计算图基础介绍

2. 具体内容

- 计算图（数据流图）：AI系统化问题 – 计算图的提出
- 计算图和自动微分：深度学习与微分 – 回顾自动微分 – 计算图表达自动微分
- 图的调度和执行：单算子调度 – 图切多设备调度 – 控制流控制
- 计算图的挑战与未来

计算图和自动微分

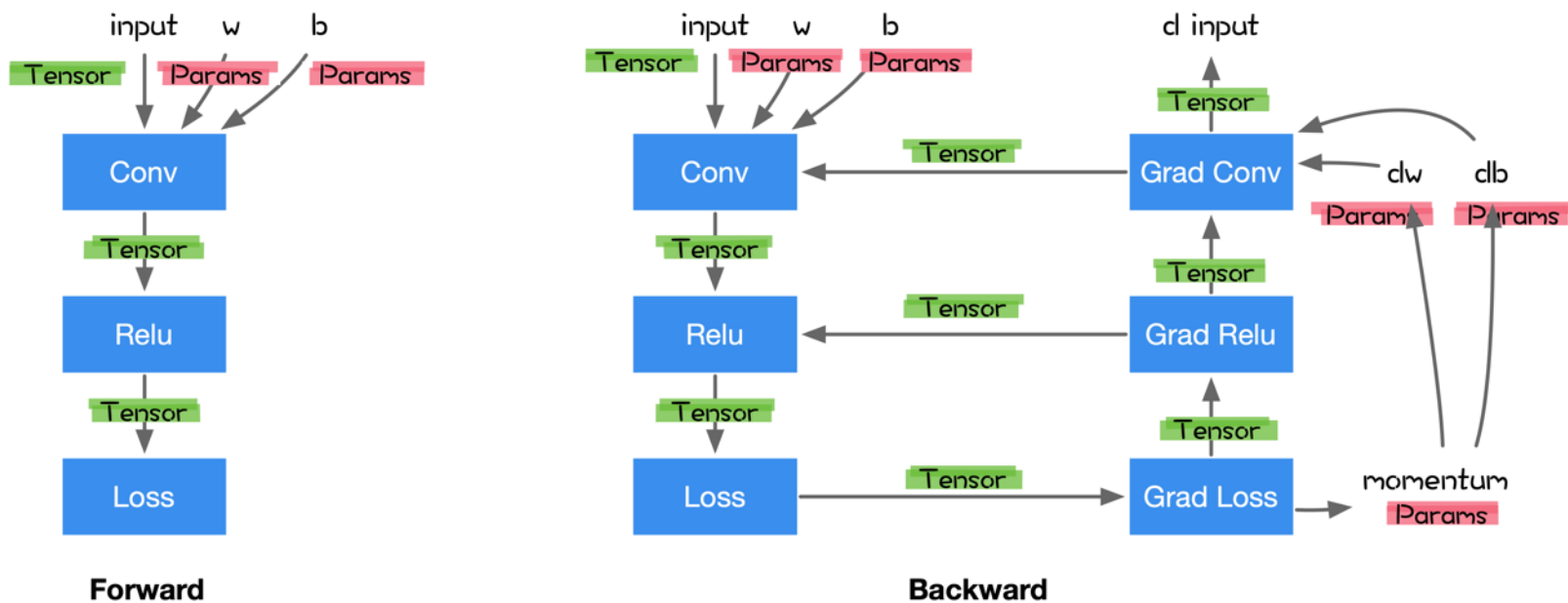
有向无环图 (DAG, Directed Acyclic Graph)



Reverse Adjoint (Derivative) Trace		
$\bar{x}_1 = \bar{v}_{-1}$		= 5.5
$\bar{x}_2 = \bar{v}_0$		= 1.716
<hr/>		
$\bar{v}_{-1} = \bar{v}_{-1} + \bar{v}_1 \frac{\partial v_1}{\partial v_{-1}}$	$= \bar{v}_{-1} + \bar{v}_1 / v_{-1}$	= 5.5
$\bar{v}_0 = \bar{v}_0 + \bar{v}_2 \frac{\partial v_2}{\partial v_0}$	$= \bar{v}_0 + \bar{v}_2 \times v_{-1}$	= 1.716
$\bar{v}_{-1} = \bar{v}_2 \frac{\partial v_2}{\partial v_{-1}}$	$= \bar{v}_2 \times v_0$	= 5
$\bar{v}_0 = \bar{v}_3 \frac{\partial v_3}{\partial v_0}$	$= \bar{v}_3 \times \cos v_0$	= -0.284
$\bar{v}_2 = \bar{v}_4 \frac{\partial v_4}{\partial v_2}$	$= \bar{v}_4 \times 1$	= 1
$\bar{v}_1 = \bar{v}_4 \frac{\partial v_4}{\partial v_1}$	$= \bar{v}_4 \times 1$	= 1
$\bar{v}_3 = \bar{v}_5 \frac{\partial v_5}{\partial v_3}$	$= \bar{v}_5 \times (-1)$	= -1
$\bar{v}_4 = \bar{v}_5 \frac{\partial v_5}{\partial v_4}$	$= \bar{v}_5 \times 1$	= 1
<hr/>		
$\bar{v}_5 = \bar{y}$	$= 1$	

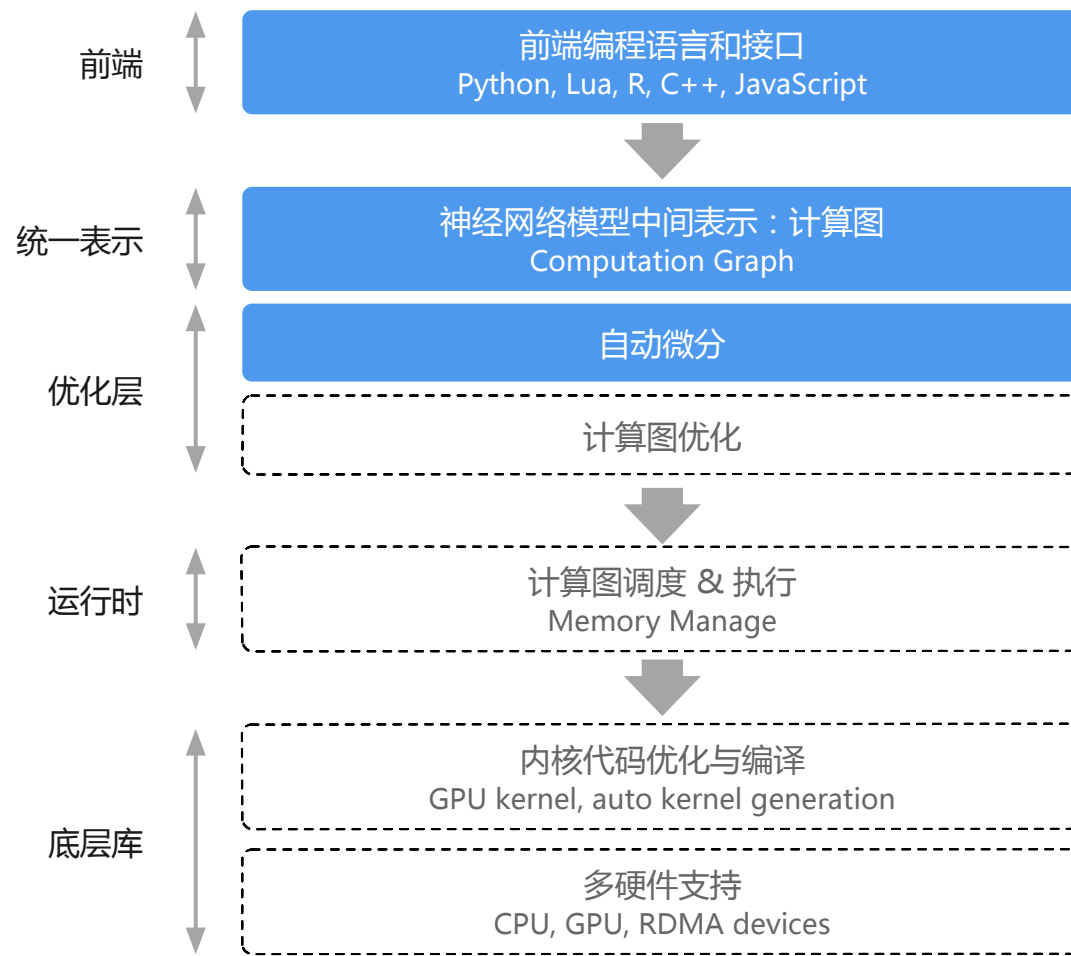
计算图的表示

- 结点：是一个后端支持的无状态的原子操作
- 边：结点之间的边上流动的是张量
- 关系：显示地表示了结点之间的Tensor 数据依赖关系



Benefit (I) : 方便底层编译优化

- 统一表示来描述神经网络训练的全过程
- 编译期可对计算过程的数据依赖进行分析：
 - 简化数据流图
 - 动态和静态内存优化
 - 预计算算子间的调度策略
 - 改善运行时Runtime性能



Benefit (II) : 分层优化 , 便于扩展

- 切分出三个解耦的优化层 :
 - 计算图优化
 - 运行时调度优化
 - 算子/内核执行优化
- 新网络模型结构/新训练算法 , 扩展步骤 :
 - 计算图层添加新算子
 - 针对不同硬件内核 , 实现计算优化
 - 注册算子和内核函数 , 运行时派发硬件执行



思考：

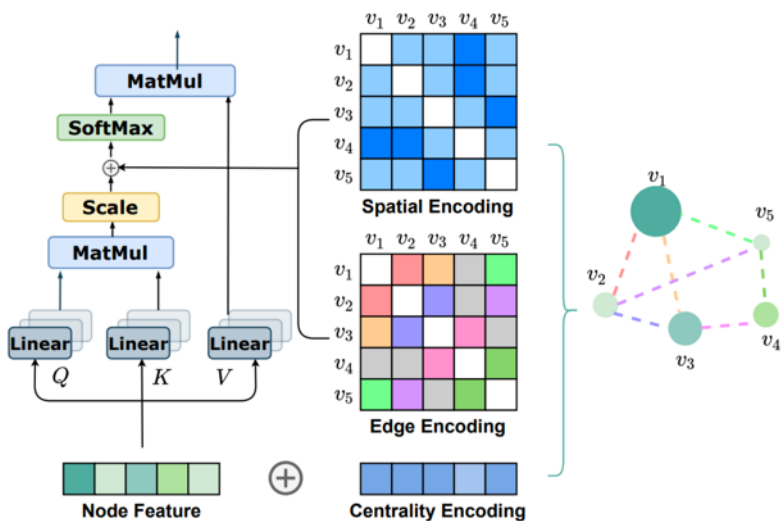
- 一定需要明确的分层解耦？
- PyTorch没有这么多层，为什么这么成功？
- 面向新的Diffusion、Transform底层优化将如何演变？
- 计算图不能解决哪些AI业务的问题？



Future Deep Dive

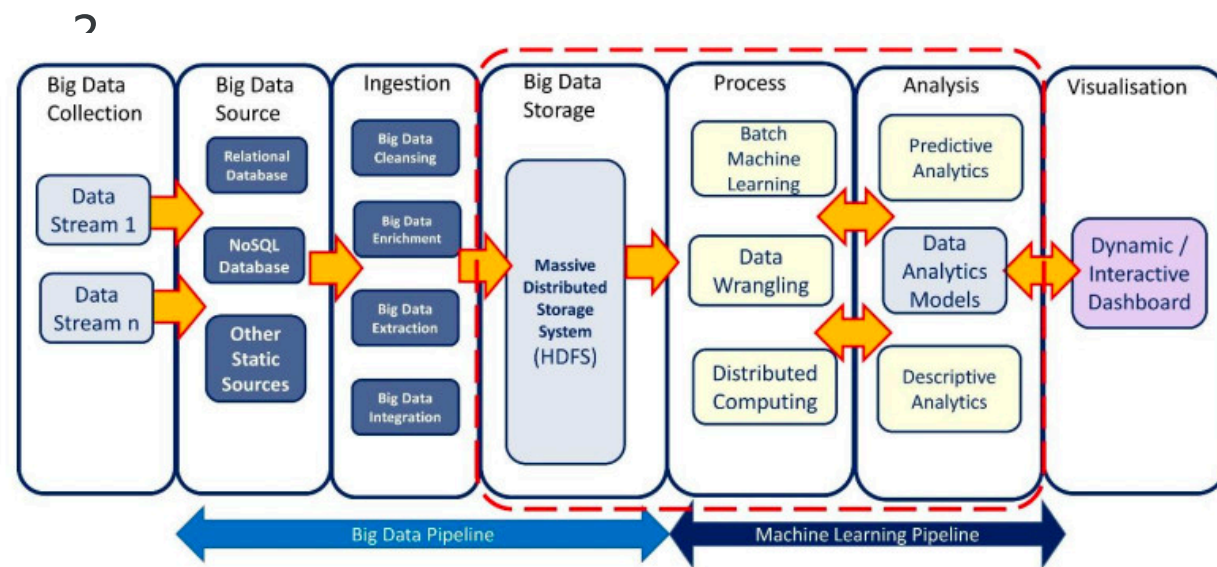
图

- 图神经网络 (GNN) 真的适合图的表示 ?
- 如何通过计算图更加高效表示图 ?



数据

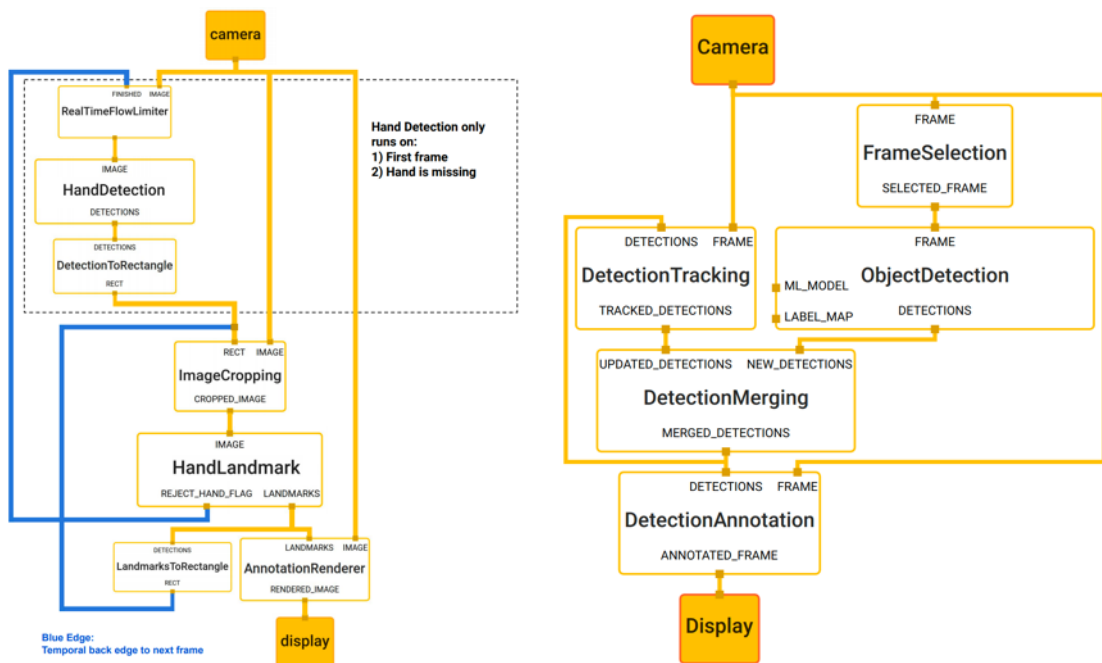
- 数据的处理(.data)如何跟计算图融合与表示 ?
- 大数据和AI融合, 计算图会产生什么新的表示



Future Deep Dive

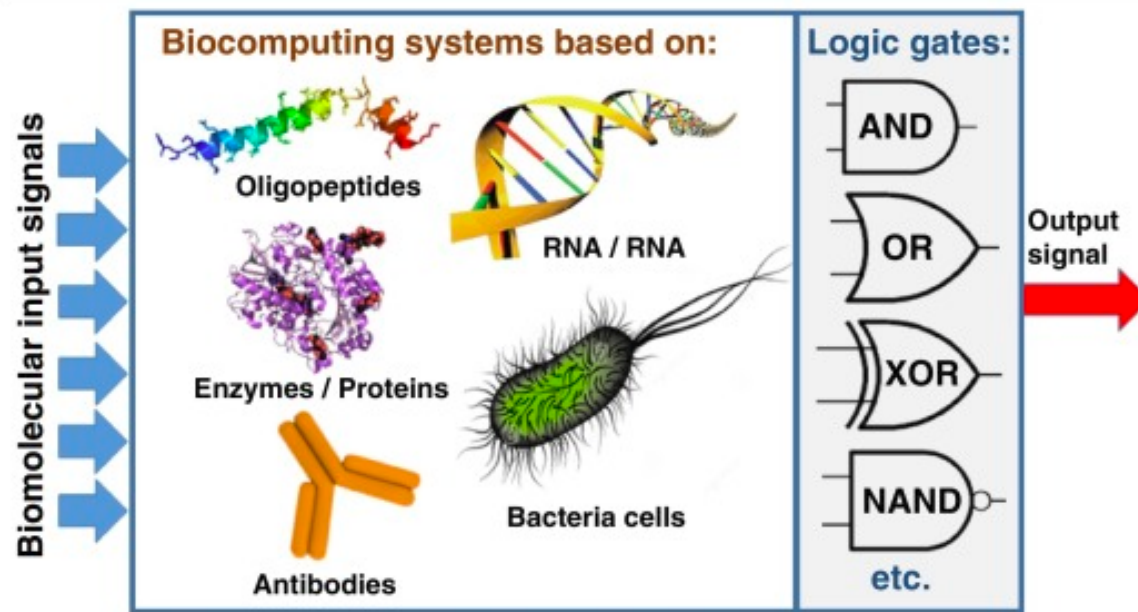
部署

- 部署场景 AI 作为流程中的一部分，能否泛化计算图解决部署流程复杂问题？



科学计算

- AI和科学计算融合，计算图能否足够表示？
- 科学计算范式，如何融合到计算图中？



Summary

1. 了解计算图和AI框架的关系
2. 展望了计算图未来与新的应用场景

