



MindSpore开源社区介绍

主办单位：

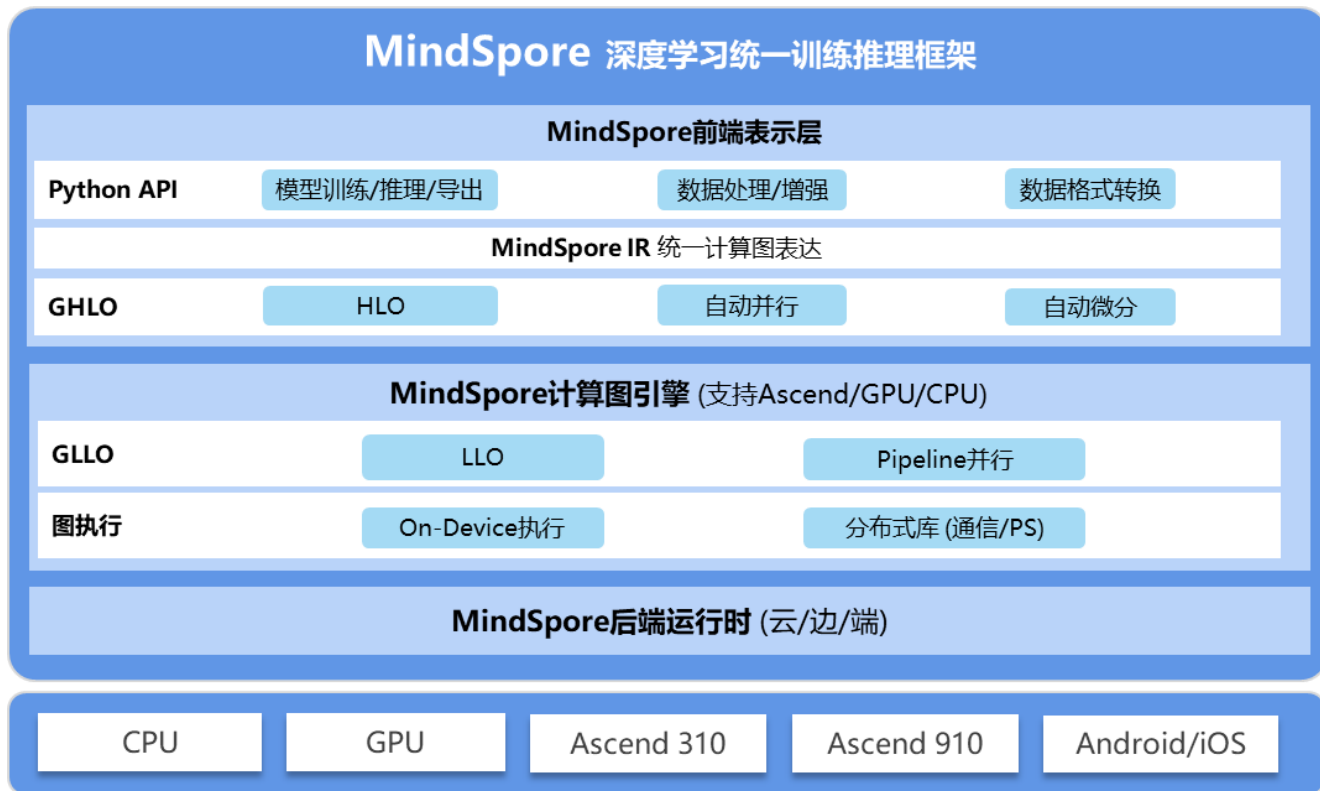


中国科学院软件研究所
Institute of Software Chinese Academy of Sciences



openEuler

MindSpore总体架构



MindSpore自动微分



```
def f(x):  
    a = x * x  
    b = x * a  
    c = a + b  
    return c
```

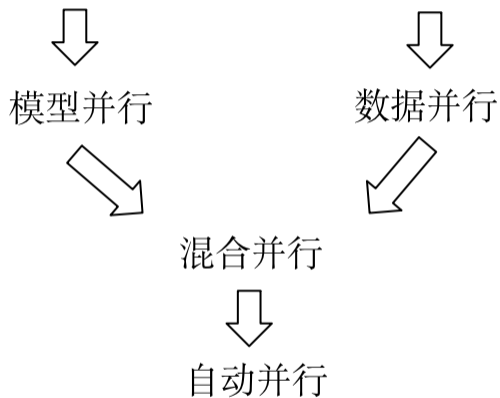
- 静态算子生成, 基于 compute表达式 (Static operator generation, based on compute expression)
- 支持高阶求导 (Support high-order differentiation)

Tangent自动微分^[1]

[1] <https://www.jiqizhixin.com/articles/2017-11-07>

MindSpore自动并行

将大规模深度学习模型在大规模数据上进行训练

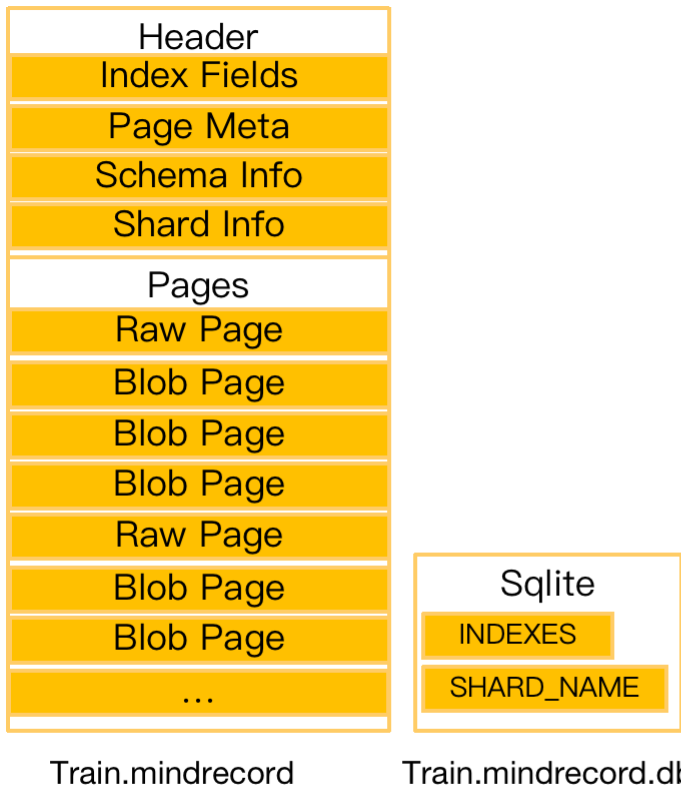


```
class DenseNet(nn.Cell):
    def __init__(self):
        super(DenseMutMulNet, self).__init__()
        self.embedding_weight = Parameter(Tensor(12288, 128))
        self.embedding = P.MatMul()
        self.fc1 = nn.Dense(128, 768, activation='relu')
        self.fc2 = nn.Dense(128, 768, activation='relu')
        self.fc3 = nn.Dense(128, 768, activation='relu')
        self.transpose = P.Transpose()
        self.matmul1 = P.MatMul()
        self.matmul2 = P.MatMul()

    def construct(self, x):
        x = self.embedding(x, self.embedding_weight)
        q = self.fc1(x)
        k = self.fc2(x)
        v = self.fc3(x)
        k = self.transpose(k, (1, 0))
        c = self.matmul1(q, k)
        s = self.matmul2(c, v)
        return s

    def train_step():
        context.set_auto_parallel_context(parallel_mode=ParallelMode.AUTO_PARALLEL)
        input = Tensor(np.ones([32, 128]).astype(np.float32))
        label = Tensor(np.zeros([32, 768]).astype(np.float32))
        net = DenseNet()
        model = Model(net, opt, loss)
        model.train(input, label)
```

MindSpore数据处理



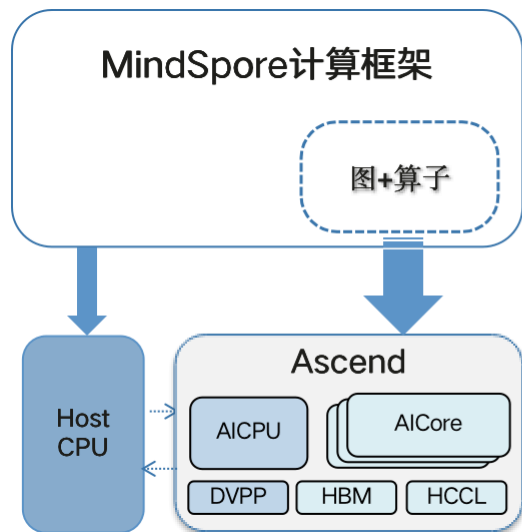
设计思路

基于不同数据自有特性实现数据的分块存储，并结合多线程技术，实现优于TFRecord的读取性能。

关键特性

- 用户自定义Schema
- 生成特定索引信息
- 零散数据 -> 数据聚合存储
- 高效MindDataset读取训练

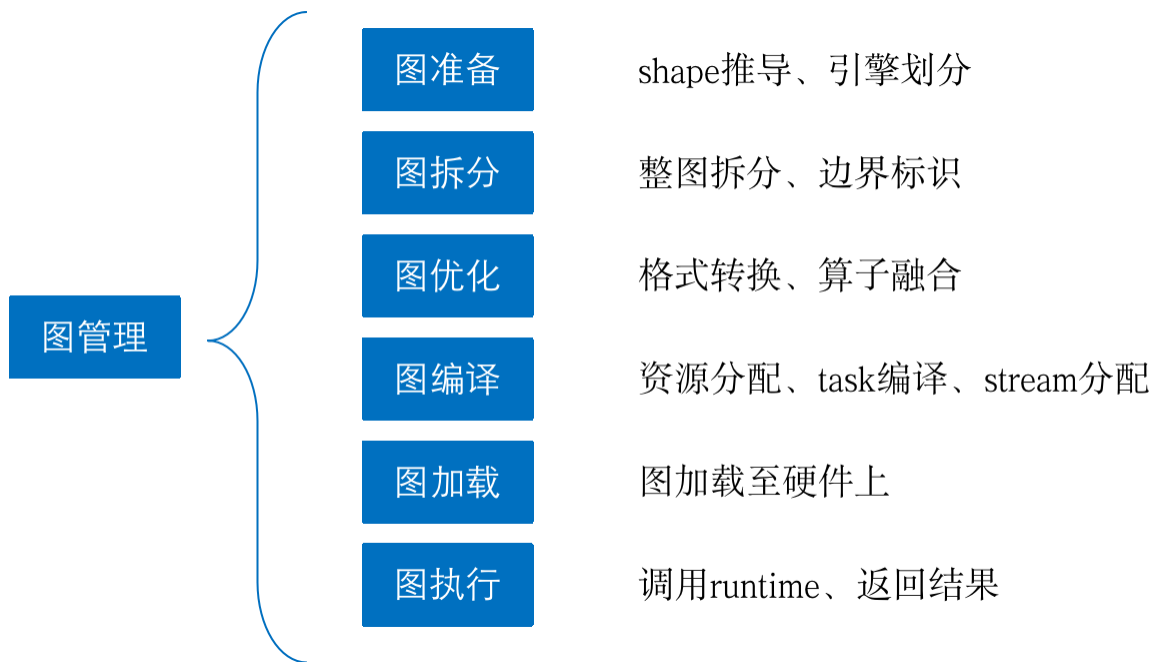
MindSpore图执行引擎



- 片上32G内存: 5G(parameter) + 26G(feature map) + 1G(HCCL)
- 多流水线并行: 6条流水线
- AICORE&带宽: 32Cores、读写带宽128GBps
- 通信协议: HCCS、PCIe4.0、RoCEv2

同时掌握硬件和软件，整体的优化空间很大

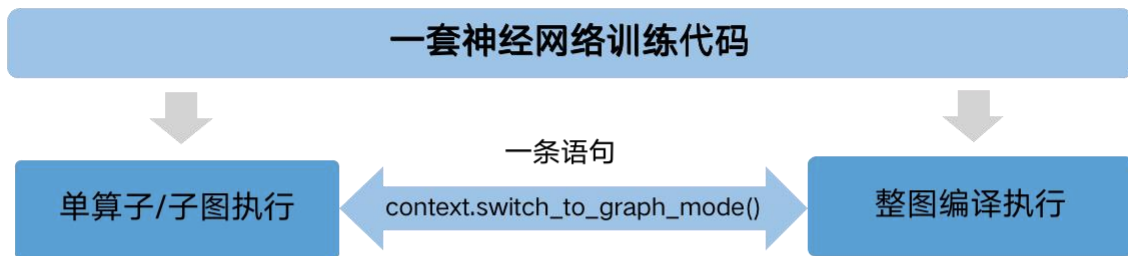
MindSpore图执行引擎



MindSpore图执行引擎

调试灵活

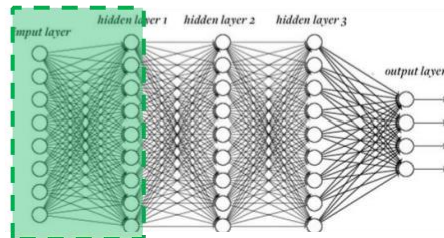
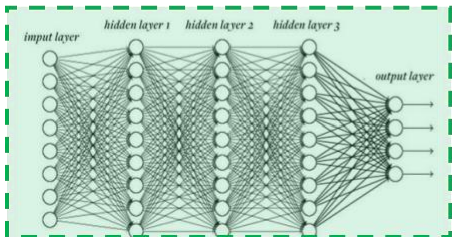
运行高效



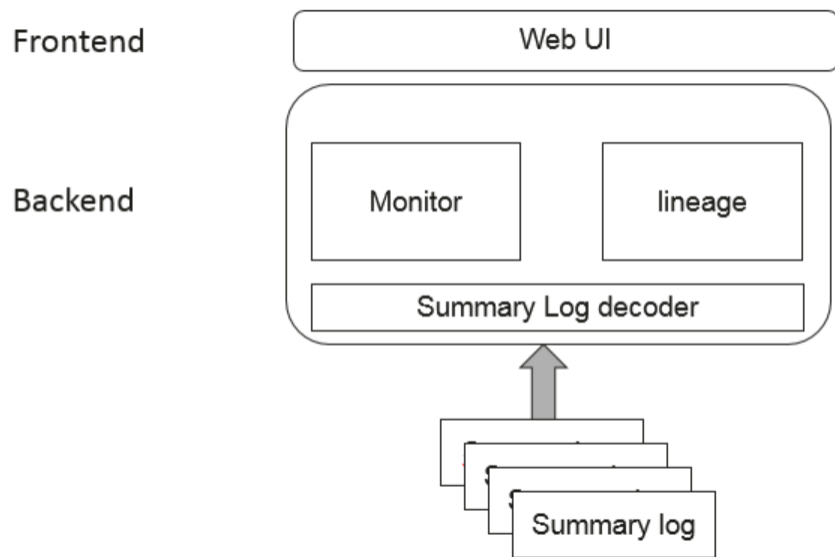
动态图：灵活的开发调试

静态图：高效的图编译优化，性能高

1行代码完成PDB调试与运行切换



MindSpore可视化工具



- MindInsight架构
 - Summary文件收集运行数据
 - 后端解析Summary文件，整理各类型数据
 - 前端对数据分类展示

MindSpore可视化工具

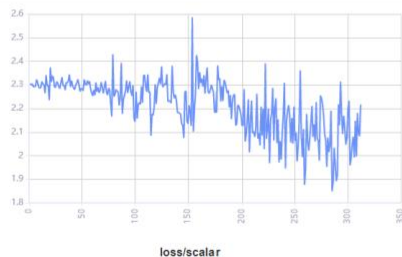
```
from mindspore.ops import functional as F
```

```
class CrossEntropyLoss(nn.Cell):
```

```
    Define loss for network
```

```
    def __init__(self):
        super(CrossEntropyLoss, self).__init__()
        self.cross_entropy = P.SoftmaxCrossEntropyWithLogits()
        self.mean = P.ReduceMean()
        self.one_hot = P.OneHot()
        self.on_value = Tensor(1.0, mstype.float32)
        self.off_value = Tensor(0.0, mstype.float32)
        self.scalar_summary = P.ScalarSummary()

    def construct(self, logits, label):
        label = self.one_hot(label, F.shape(logits)[1], self.on_value, self.off_value)
        loss = self.cross_entropy(logits, label)[0]
        loss = self.mean(loss, (-1,))
        self.scalar_summary("loss", loss)
        return loss
```

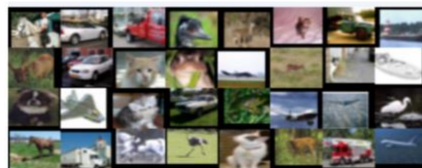


```
class AlexNet(nn.Cell):
```

```
    Alexnet
```

```
    def __init__(self, num_classes=1001):
        super(AlexNet, self).__init__()
        self.batch_size = 32
        self.conv1 = conv(3, 96, 11, stride=4)
        self.conv2 = conv(96, 256, 5, pad_mode="same")
        self.conv3 = conv(256, 384, 3, pad_mode="same")
        self.conv4 = conv(384, 384, 3, pad_mode="same")
        self.conv5 = conv(384, 96, 3, pad_mode="same")
        self.relu = nn.ReLU()
        # self.max_pool2d = nn.MaxPool2d(kernel_size=3, stride=2)
        self.max_pool2d = P.MaxPool(ksize=3, strides=2)
        self.reshape = P.Reshape()
        self.fc1 = fc_with_initialize(6*6*96, 4096)
        self.fc2 = fc_with_initialize(4096, 4096)
        self.fc3 = fc_with_initialize(4096, num_classes)
        self.image_summary = P.ImageSummary()

    def construct(self, x):
        self.image_summary("image", x)
        x = self.conv1(x)
        x = self.relu(x)
```



image/image
./summary_alexnet_200430_170047

步骤: 312

2020/4/30 下午5:07:08

图像可视

```
class Momentum(Optimizer):
```

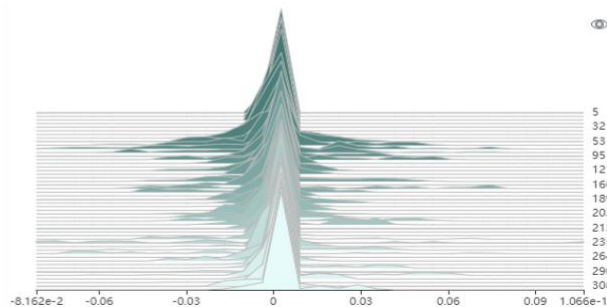
```
    ... ..
```

```
    # Initialize ScalarSummary.
    self.sm_scalar = P.ScalarSummary()

    # # prepare summary operator
    self.histogram_summary = P.HistogramSummary()
    # # prepare tags
    self.weight_names = [param.name for param in self.parameters]
    self.gradient_names = [param.name + ".gradient" for param in self.parameters]
    self.param_count = len(self.parameters)
```

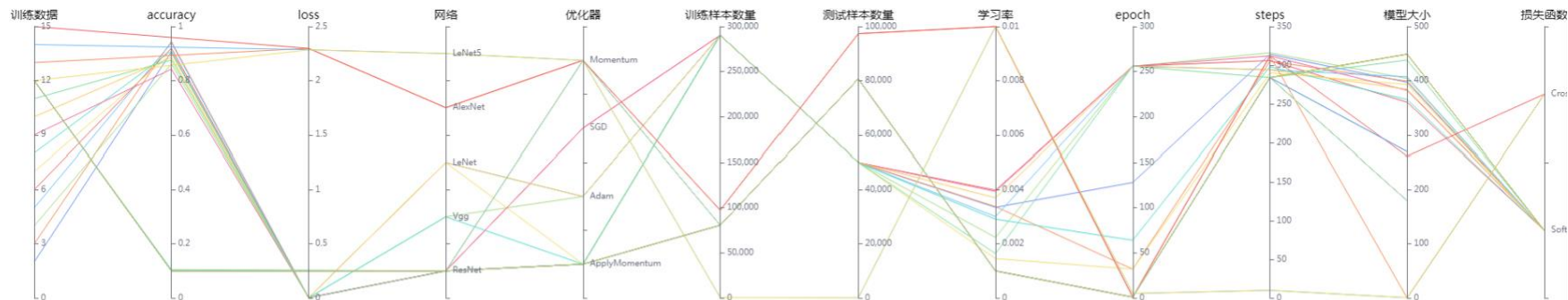
```
    def construct(self, grads):
        # record weights
        for i in range(self.param_count):
            self.histogram_summary(self.weight_names[i], self.params[i])

        # record gradients
        for i in range(self.param_count):
            self.histogram_summary(self.gradient_names[i], grads[i])
```

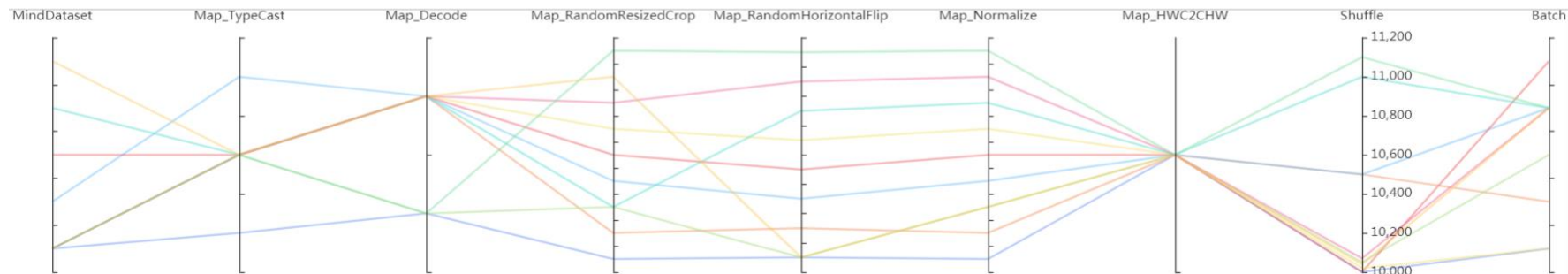


直方图可视

MindSpore可视化工具



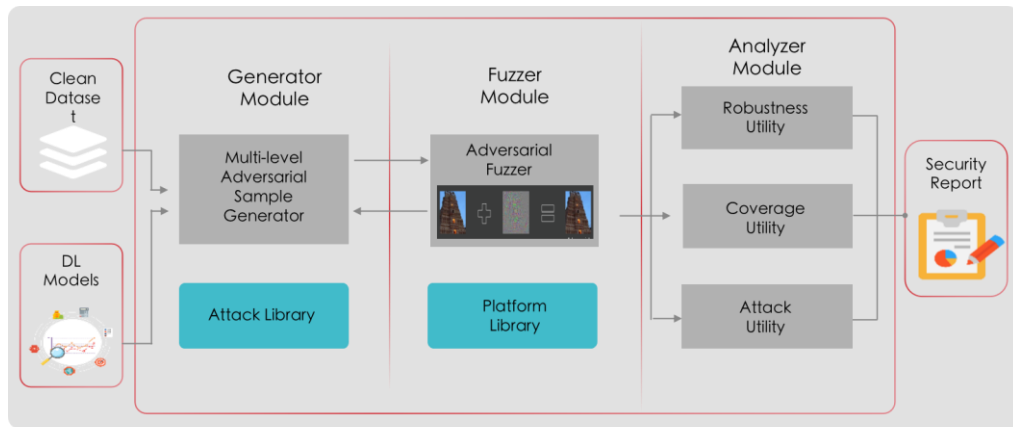
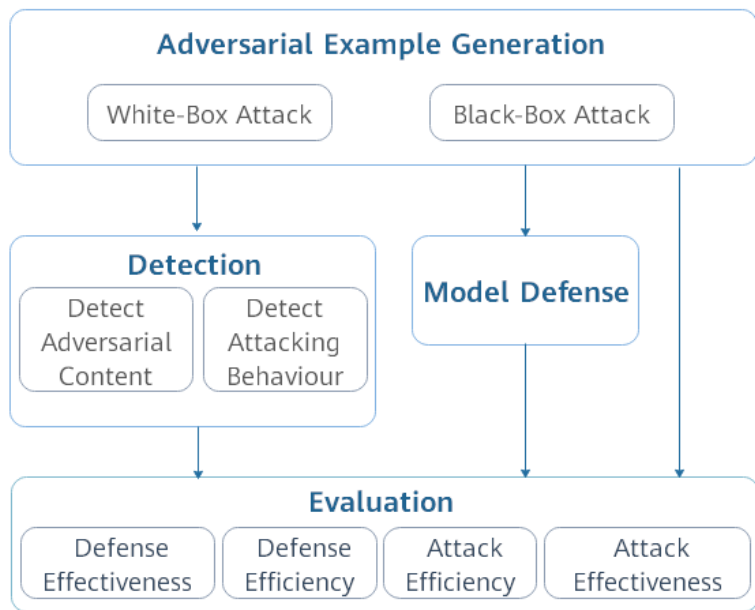
- 直观查看影响模型效果的参数组合，以辅助分析决策和模型复现；



- 发现数据算子变化对训练的影响

MindSpore模型评估工具

MindArmour



AI模型测试：通过fuzzing技术给出模型安全性分析

MindSpore暑期2020题目

基于MindSpore的深度学习算法开发

项目描述：MindSpore是华为自研的深度学习框架，采用基于源码转换的自动微分，支持自动并行、深度图优化。MindSpore原生支持昇腾芯片，针对昇腾芯片架构特点，采用Pipeline并行、整图下沉等功能，可以达到很高的执行性能。目前MindSpore还处于发展阶段，本项目向大家介绍MindSpore特性及API使用，希望大家基于MindSpore进行深度学习算法开发，完成某项指定挑战。

项目难度：中

项目社区导师：王俊

导师联系方式：wangjun289@huawei.com

项目产出要求：

完成下列三项任务之一

1. 提交基于MindSpore框架的深度学习模型以完成一项任务（会提供若干参考题目，但不限制选题内容）
2. 开发指定MindSpore算子
3. 解决指定MindSpore相关issue

最后制作ppt进行展示

项目技术要求：

- 基础Python使用
- 一定深度学习基础

相关的开源软件仓库列表：

- <https://www.mindspore.cn>
- <http://gitee.com/mindspore/mindspore>

MindSpore暑期2020题目

深度学习模型计算图相同子结构的识别和展示

项目描述：MindSpore是华为自研的深度学习框架，其中的计算图模式是一种业界主流的用来进行数据传递与计算的形式。计算图主要包含节点和有向边，节点表示计算和控制操作，边表示数据的流向和控制等关系。计算图的高效合理展示，有助于用户更好的理解模型结构、发现和调试模型训练过程中出现的问题。然而，大型深度学习模型往往有着复杂的计算图结构，包含有成千上万个节点和更多的边。在这些点和边之中，包含有许多结构相同或高度相似的子结构，这些子结构不仅从图的拓扑结构上，甚至从深度学习语义上具有高度的相似性。快速识别大型计算图中上述的相同子结构，能够支持后续用收折、重叠等方式大幅减少页面中同时呈现的节点和边的数目，从而大幅改善计算图的展示效果。

项目难度：中

项目社区导师：高寒

导师联系方式：gaohan19@huawei.com

项目产出要求：

完成一个相同图结构识别工具，其中

- 输入：MindSpore的计算图文件（以 `protobuf` 格式序列化存储）
- 输出：相同的子结构（合理形式即可，如json等，内含字典，字典的键为编号，值为一个List，其中每个元素又为一个List，包含了相同子结构点和边的连接关系）

最后制作ppt进行展示

项目技术要求：

- 基础Python使用
- 一定的深度学习基础

问答环节

MindSpore开源社区：

<https://www.mindspore.cn>

MindSpore代码：

<https://gitee.com/mindspore/mindspore>

MindSpore暑期2020题目：

<https://gitee.com/mindspore/community/tree/master/events/summer2020>

微信可添加小助手：

Mindspore0328

QQ讨论群：

871543426

B站：

526894060

抖音号： 592553310

感谢



欢迎关注暑期2020活动官网