MindSpore

MindSpore 功能调试实践

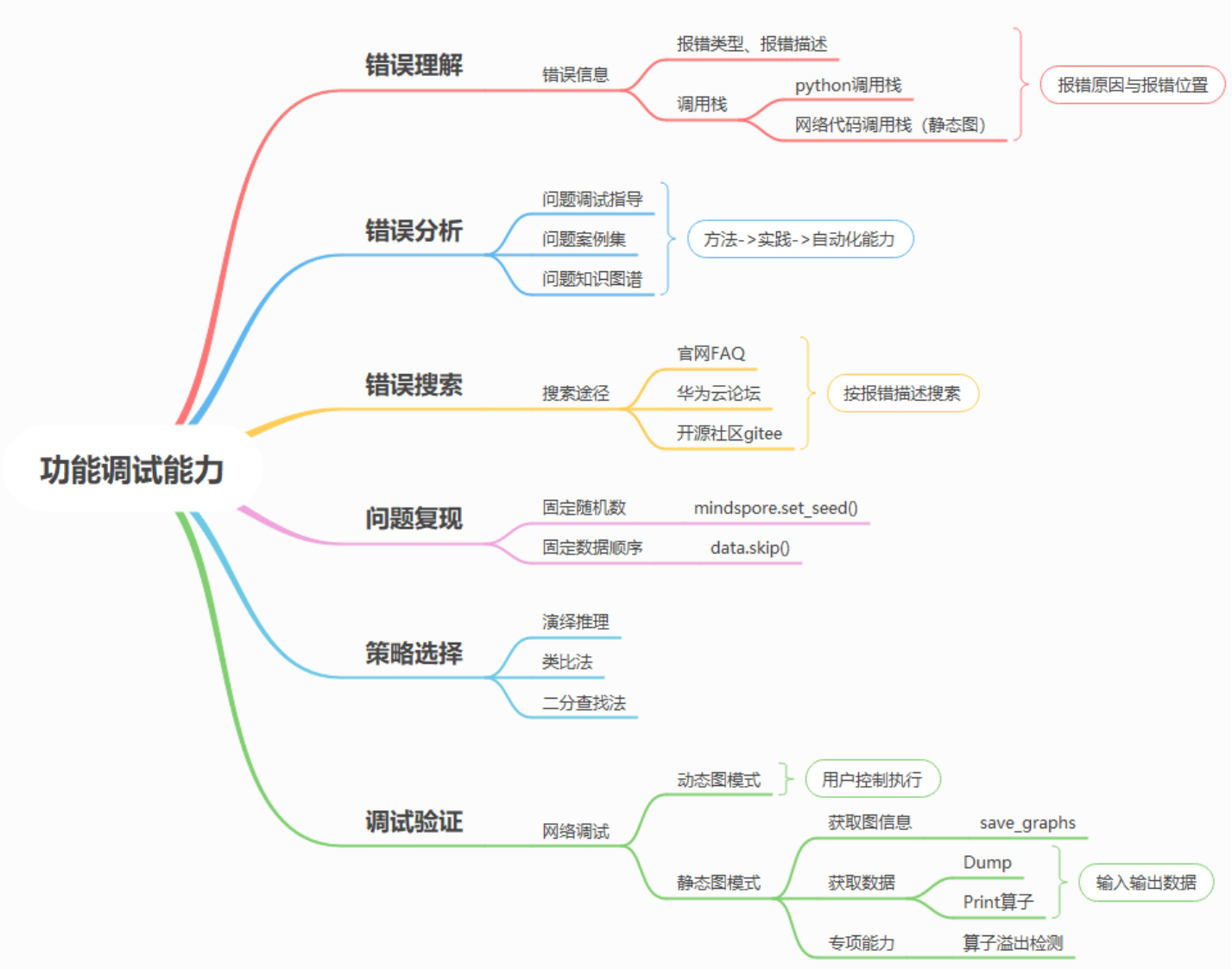
作者：Louie
时间：2022-5-28

目录

contents

MindSpore

MindSpore功能调试的能力



目标：报错可理解、问题可复现 、过程可跟踪、案例覆盖全。

MindSpore功能调试方法

MindSpore

1、静态图编译报错信息

```
1 [EXCEPTION] ANALYZER(31946,7f6f03941740,python):2021-09-18-15:10:49.094.863 [mindspore/ccsrc/pipeline/jit/static_analy
2 FunctionGraph ID : func.18
3 NodeInfo: In file test.py(19)
4     def func(x, y):
5
6 Traceback (most recent call last):
7   File "test.py", line 31, in <module>
8     out = net(input1, input2)
9   File "/home/workspace/mindspore/mindspore/nn/cell.py", line 404, in __call__
10    out = self.compile_and_run(*inputs)
11  File "/home/workspace/mindspore/mindspore/nn/cell.py", line 682, in compile_and_run
12    self.compile(*inputs)
13  File "/home/workspace/mindspore/mindspore/nn/cell.py", line 669, in compile
14    _cell_graph_executor.compile(self, *inputs, phase=self.phase, auto_parallel_mode=self._auto_parallel_mode)
15  File "/home/workspace/mindspore/mindspore/common/api.py", line 542, in compile
16    result = self._graph_executor.compile(obj, args_list, phase, use_vm, self.queue_name)
17 TypeError: mindspore/ccsrc/pipeline/jit/static_analysis/stack_frame.cc:85 DoJump] The parameters number of the functi
18 FunctionGraph ID : func.18
19 NodeInfo: In file test.py(19)
20     def func(x, y):
21
22 The function call stack (See file '/home/workspace/mindspore/rank_0/om/analyze_fail.dat' for more details):
23 # 0 In file test.py(26)
24         return c
25         ^
26 # 1 In file test.py(25)
27         c = self.mul(b, self.func(a, a, b))
28                     ^
```

Python调用栈

错误描述

报错代码行

2、静态图执行的报错信息

```
2022-02-14 13:15:14,879:INFO:epoch[0], iter[7376], loss:4.341205, fps:35.98 imgs/sec, lr:0.00079906883
2022-02-14 13:15:15,757:INFO:epoch[0], iter[7377], loss:6.163947, fps:35.32 imgs/sec, lr:0.00079917704
2022-02-14 13:15:16,662:INFO:epoch[0], iter[7378], loss:5.126837, fps:35.16 imgs/sec, lr:0.00079928525
2022-02-14 13:15:17,602:INFO:epoch[0], iter[7379], loss:5.011183, fps:34.35 imgs/sec, lr:0.0007993934
2022-02-14 13:15:18,516:INFO:epoch[0], iter[7380], loss:5.311672, fps:33.19 imgs/sec, lr:0.0007995016
2022-02-14 13:15:19,382:INFO:epoch[0], iter[7381], loss:5.209589, fps:35.94 imgs/sec, lr:0.0007996098
[EXCEPTION] GE(55333,7fcdf4ff9700,python):2022-02-14-13:15:20.262.133 [mindspore/ccsrc/runtime/device/ascend/ge_runtime/runtime_model.cc:233] Run] Call rt api rtStreamSynchronize failed, ret:
[WARNING] DEVICE(55333,7fcdf4ff9700,python):2022-02-14-13:15:20.262.747 [mindspore/ccsrc/runtime/device/ascend/ascend_kernel_runtime.cc:667] GetDumpPath] MS_OM_PATH is null, so dump to process
, as ./rank_id/node_dump/...
[ERROR] DEVICE(55333,7fcdf4ff9700,python):2022-02-14-13:15:20.262.851 [mindspore/ccsrc/runtime/device/ascend/ascend_kernel_runtime.cc:684] DumpTaskExceptionInfo] Task fail infos task_id: 221,
18, tid: 68322, device_id: 3, retcode: 507011 ( model execute failed)
[ERROR] DEVICE(55333,7fcdf4ff9700,python):2022-02-14-13:15:20.292.526 [mindspore/ccsrc/runtime/device/ascend/ascend_kernel_runtime.cc:693] DumpTaskExceptionInfo] Dump node (Default/optimizer-S
75) task error input/output data to: ./rank_0/node_dump trace:
In file /root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/nn/optim/optimizer.py(729)/    return op_mul(grad, F.cast(scale, F.dtype(grad)))/
Corresponding forward node candidate:


[EXCEPTION] SESSION(55333,7fcdf4ff9700,python):2022-02-14-13:15:20.308.435 [mindspore/ccsrc/backend/session/ascend_session.cc:1551] Execute] run task error!
EZ9999: Inner Error!
        The device(0), core list[0-0], error code is:[FUNC:ProcessCoreErrorInfo][FILE:device_error_proc.cc][LINE:420]
        coreId( 0):       0x800000    [FUNC:ProcessCoreErrorInfo][FILE:device_error_proc.cc][LINE:431]
        Aicore kernel execute failed, device_id=3, stream_id=18, report_stream_id=12, task_id=221, fault kernel_name=mul_13238695283139552833_3__kernel0, func_name=mul_13238695283139552833_3_
ogram id=559, hash=14948142473620195876[FUNC:GetError][FILE:stream.cc][LINE:711]
        Stream synchronize failed, stream = 0x7fcaf005cda0[FUNC:StreamSynchronize][FILE:logger.cc][LINE:285]
        rtStreamSynchronize execute failed, reason=[the model stream execute failed][FUNC:ReportFuncErrorReason][FILE:error_message_manage.cc][LINE:41]

[ERROR] ME(55333:140567594854208,MainProcess):2022-02-14-13:15:20.312.464 [mindspore/dataset/engine/datasets.py:2686] Uncaught exception:
Traceback (most recent call last):
  File "train-test-lr-new.py", line 302, in <module>
    run_train(args_wrapper)
  File "/disk0/txy/88.9/sunzhongqian/M2Det-graph-modify/src/moxing_adapter.py", line 109, in wrapped_func
    run_func(args_wrapper,*args, **kwargs)
  File "train-test-lr-new.py", line 274, in run_train
    loss = net(images, loc_t, conf_t)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/nn/cell.py", line 407, in __call__
    out = self.compile_and_run(*inputs)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/nn/cell.py", line 750, in compile_and_run
    return _cell_graph_executor(self, *new_inputs, phase=self.phase)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/common/api.py", line 630, in __call__
    return self.run(obj, *args, phase=phase)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/common/api.py", line 658, in run
    return self._exec_pip(obj, *args, phase=phase_real)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/common/api.py", line 78, in wrapper
    results = fn(*arg, **kwargs)
  File "/root/miniconda3/envs/sunzq/lib/python3.7/site-packages/mindspore/common/api.py", line 641, in _exec_pip
    return self._graph_executor(args_list, phase)
RuntimeError: mindspore/ccsrc/backend/session/ascend_session.cc:1551 Execute] run task error!

#
```

> Traceback后面的内容是重点报错信息。包括Python调用栈和错误描述。
> The function call stack 是网络代码调用栈，包含报错代码行。

> Inner Error信息和第一个Error日志是关键信息；
> 报错算子不一定是问题引入点。

MindSpore功能调试方法

MindSpore

1、静态图编译错误分析

```
19      def func(x, y):
20          return self.div(x, y)
21
22      def construct(self, x, y):
23          a = self.sub(x, 1)
24          b = self.add(a, y)
25          c = self.mul(b, self.func(a, a, b))
26          return c
```

```
17 TypeError: mindspore/ccsrc/pipeline/jit/static_analy
18 FunctionGraph ID : func.18
19 NodeInfo: In file test.py(19)
20      def func(x, y):
```

```
1 # [No.1] construct_wrapper.0
2 # In file test.py(22)/    def construct(self, x, y):/
3 funcgraph fg_0(
4         %para1 : Tensor(F32)[]    # x
5         , %para2 : Tensor(F32)[]    # y
6     ) {
7
8 #------------------------> 0
9     %1 = FuncGraph::fg_3(%para1, %para2)    #(Tensor(F32)[], Tensor(F32)[])    # fg_3=construct.3 #scope: Default
10      # In file test.py(26)/        return c/#[CNode]2
11    Primitive::Return{prim_type=1}(%1)    #(Undefined) #scope: Default
12      # In file test.py(26)/        return c/#[CNode]1
13 }
14 # order:
15 #   1: construct_wrapper.0:[CNode]2{[0]: ValueNode<FuncGraph> construct.3, [1]: x, [2]: y}
16 #   2: construct_wrapper.0:[CNode]1{[0]: ValueNode<Primitive> Return, [1]: [CNode]2}
17
18
19 # [No.2] construct.3
20 # In file test.py(22)/    def construct(self, x, y):/
21 funcgraph fg_3(
22        %para3 : Tensor(F32)[]    # x
23        , %para4 : Tensor(F32)[]    # y
24    ) {
25    %1 : Tensor(F32)[] = DoSignaturePrimitive::S-Prim-Sub{prim_type=1}[input_names=["x", "y"], output_names=["output"]](%para3, I64(1))    #(Tensor(F32)[], I64) #scope: Default
26      # In file test.py(23)/        a = self.sub(x, 1)/#a
27    %2 : Tensor(F32)[] = DoSignaturePrimitive::S-Prim-Add{prim_type=1}[input_names=["x", "y"], output_names=["output"]](%1, %para4)    #(Tensor(F32)[], Tensor(F32)[]) #scope: Default
28      # In file test.py(24)/        b = self.add(a, y)/#b
29
30 #------------------------> 1
31    %3 = FuncGraph::fg_18(%1, %1, %2)    #(Tensor(F32)[], Tensor(F32)[], Tensor(F32)[])    # fg_18=func.18 #scope: Default
32      # In file test.py(25)/        c = self.mul(b, self.func(a, a, b))/#[CNode]5
33    %4 = DoSignaturePrimitive::S-Prim-Mul{prim_type=1}[input_names=["x", "y"], output_names=["output"]](%2, %3)    #(Tensor(F32)[], Undefined) #scope: Default
34      # In file test.py(25)/        c = self.mul(b, self.func(a, a, b))/#c
35    Primitive::Return{prim_type=1}(%4)    #(Undefined) #scope: Default
36      # In file test.py(26)/        return c/#[CNode]4
37 }
```

MindSpore功能调试方法

MindSpore

2、问题案例分析

1. 系统环境

Hardware Environment(Ascend/GPU/CPU): ALL
Software Environment:
MindSpore version (source or binary): 1.6.0 & Earlier versions
Python version (e.g., Python 3.7.5): 3.7.6
OS platform and distribution (e.g., Linux Ubuntu 16.04): Ubuntu
GCC/Compiler version (if compiled from source):

2. python代码样例

```
from mindspore.nn import Cell

class Net(Cell):
    def construct(x):
        return x

net = Net()
net(2)
```

3. 报错信息

Traceback (most recent call last):
File "error_map/01/param_not_match.py", line 10, in
net(2)
File "/usr/local/python3.7/lib/python3.7/site-packages/mindspore/nn/cell.py", line 469, in **call**
self._check_construct_args(*args, **kwargs)
File "/usr/local/python3.7/lib/python3.7/site-packages/mindspore/nn/cell.py", line 400, in
_check_construct_args
f"The function construct needs {positional_args} positional argument and {default_args} default "
TypeError: The function construct needs 0 positional argument and 0 default argument, but provided

4. 原因分析

首先查看报错信息“TypeError: The function construct needs 0 positional argument and 0 default argument, but provided 1”

报错信息直译为：函数 `construct` 需要0个位置参数和0个默认参数，但提供了1个

为了理解报错信息的含义，需要解释两个名词 `positional argument` 和 `default argument` 。要解释这两个名词首先要理解什么是argument(调用时输入的实参，有具体的值)，说argument就必须要说Parameter(函数定义时声明

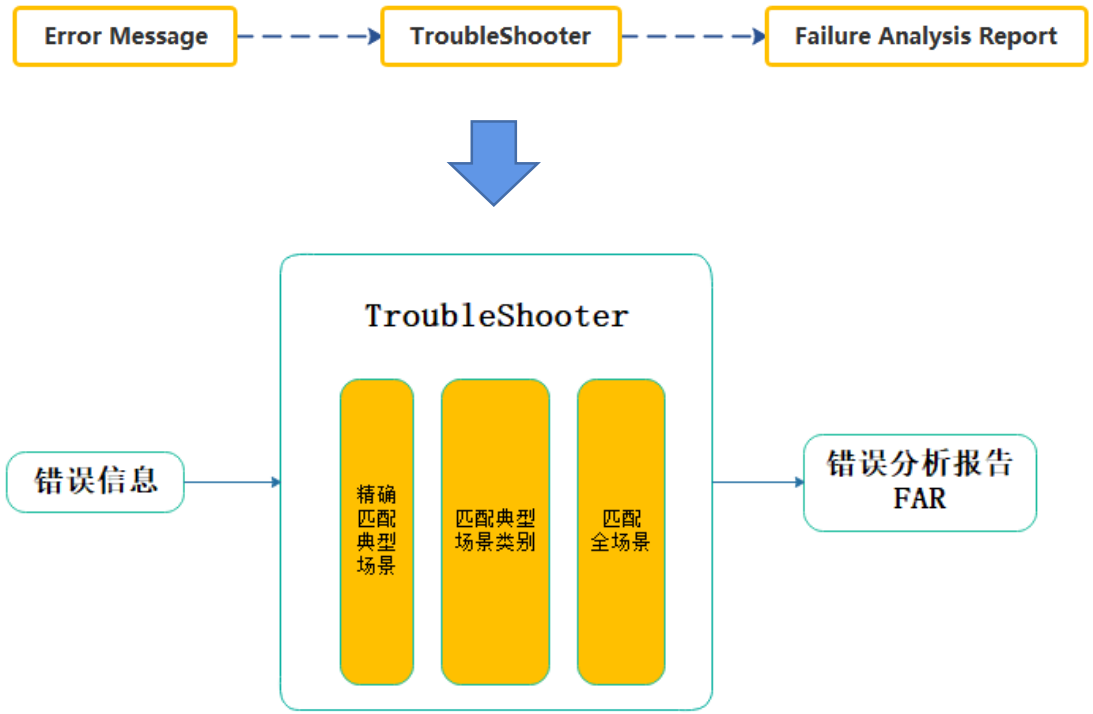
5. 解决方法

通过上一章节的分析可以看出 `def construct(x):` 的定义是错误的，缺少了 `self` 入参，其正确定义为：`def construct(self, x):`

6. 平台改进点

报错信息上尽量与原生python保持一致，方便开发者进行理解。

3、问题知识图谱：基于报错信息自动识别分析，输出错误分析报告

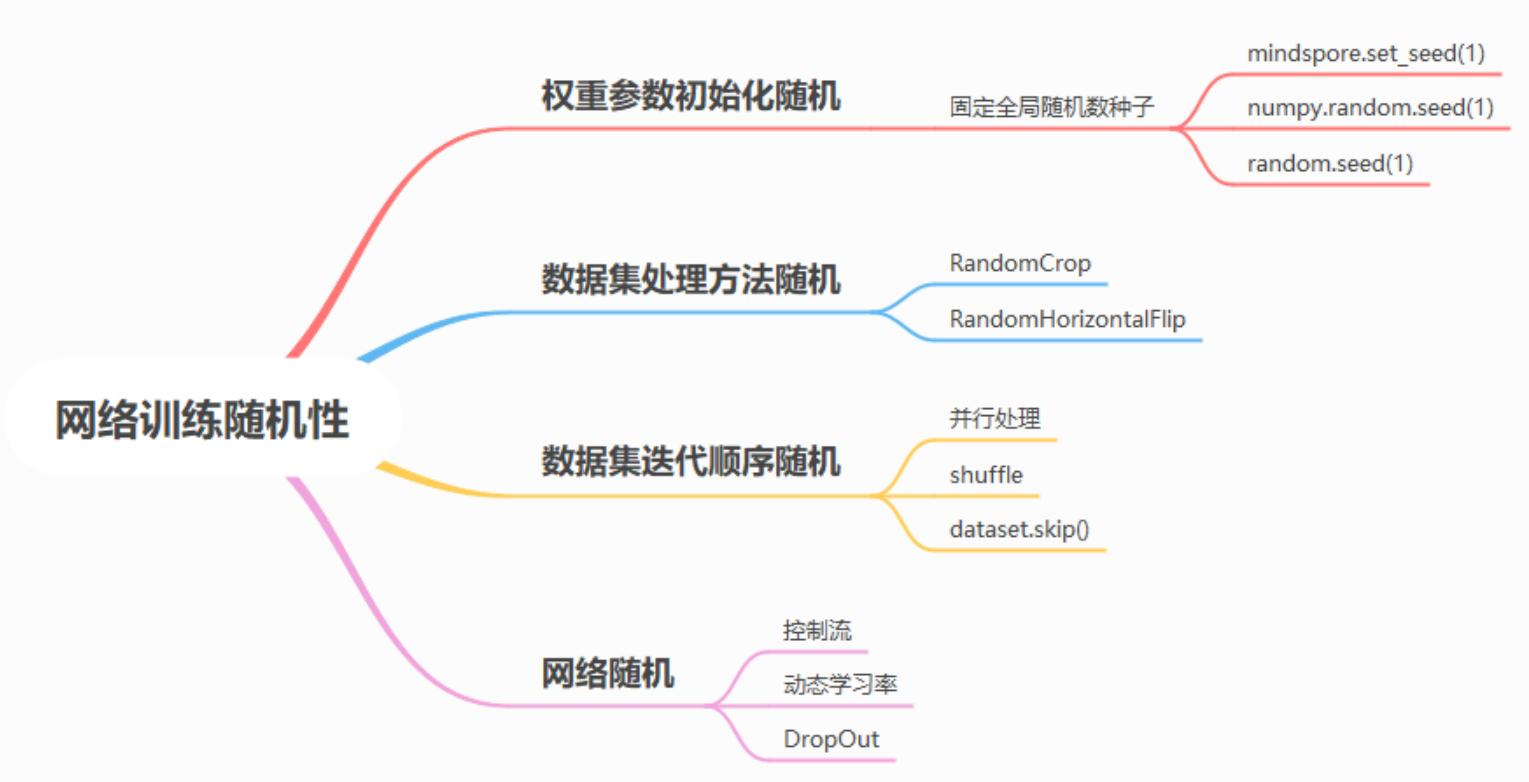


多层次匹配，报错全场景覆盖。

MindSpore功能调试方法

MindSpore

计算图执行，因随机性输入，存在问题难复现场景。



验证随机性是否固定的方法：使用$numpy.allclose$方法。

1） 两次运行脚本，首个迭代loss值满足$numpy.allclose(l_1, l_2) = True$，说明网络正向传播随机性得到固定。
2）两次运行脚本，第二个迭代loss值满足$numpy.allclose(l_1', l_2') = True$，说明网络正向和反向传播随机性都得到固定。

其中 $numpy.allclose$ 的参数： ***rtol=1e-03, atol=1e-03***

MindSpore功能调试方法

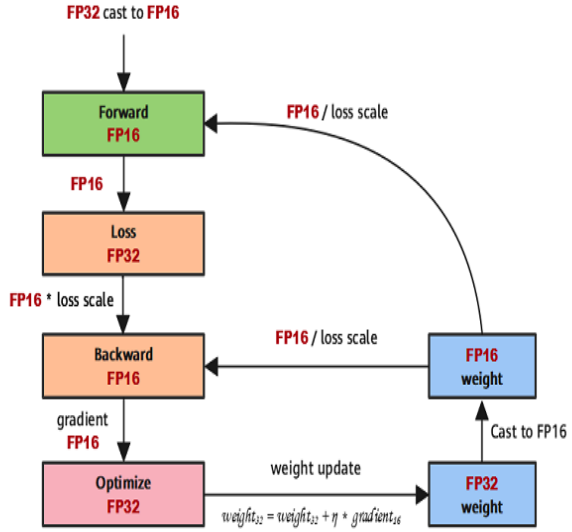
MindSpore

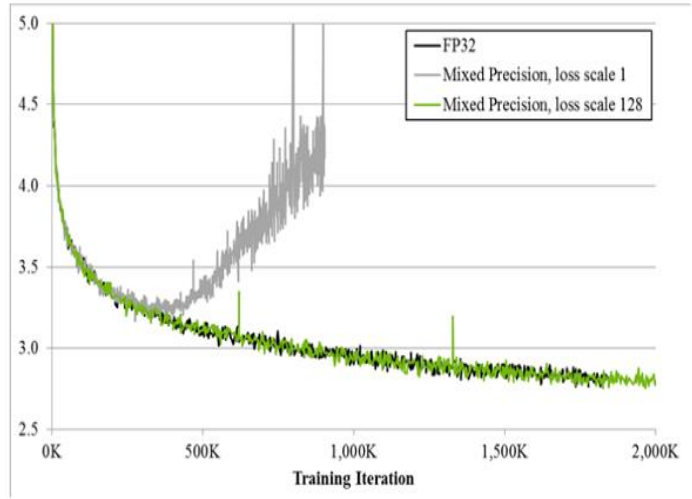
1、演绎推理

演绎推理是基于错误的因果传播链对问题进行分析。

2、类比法

类比法可以常用于分析混合精度相关问题。





使用FP16替换FP32会出现上溢和下溢的情况。而在深度学习中，梯度会比权重值更加小，往往容易出现下溢情况。

Loss Scale损失缩放，正是为了解决FP16类型数据下溢问题。

MindSpore功能调试方法

MindSpore

静态图模式调试：获取图信息、获取数据

1、保存图IR

1 配置方法

```
context.set_context(save_graphs=True, save_graphs_path="path/to/ir/files")
```

2 保存结果

```
.
├──00_parse_0000.ir
├──00_parse_0001.dat
├──00_parse_0002.dot
├──01_symbol_resolve_0003.ir
├──01_symbol_resolve_0004.dat
├──01_symbol_resolve_0005.dot
├──02_combine_like_graphs_0006.ir
├──02_combine_like_graphs_0007.dat
├──02_combine_like_graphs_0008.dot
├──03_inference_opt_prepare_0009.ir
├──03_inference_opt_prepare_0010.dat
├──03_inference_opt_prepare_0011.dot
├──04_abstract_specialize_0012.ir
├──04_abstract_specialize_0013.dat
├──04_abstract_specialize_0014.dot
...
```

3 计算图表示的含义

- **parse阶段**负责解析入口函数，该阶段会初步生成MindIR，该阶段仅仅解析了顶层Cell的图信息；
- **symbol_resolve阶段**负责进一步解析入口函数，主要是递归解析入口函数直接或间接引用到的其他函数和对象。如果使用了尚不支持的语法，一般会在此阶段出错；
- **abstract_specialize阶段**根据输入信息推导出IR中所有节点的数据类型和形状信息。该阶段可查看IR中具体算子的形状或数据类型。

2、数据dump

1 同步Dump：
在网络训练过程中每个step执行结束后， Host侧发起Dump动作，从Device上拷贝算子地址里面的数据到Host，并保存文件。

2 异步Dump：
专门针对Ascend整图下沉而开发的功能，可以一边执行算子一边dump数据，一个算子执行结束后立即dump数据，此时网络训练的速度会比较慢。

3 分析过程：
① 从脚本中找到对应算子；
② 利用算子名称找到Dump数据；
③ 构建单算子案例，使用Dump数据进行分析验证。

4 适用场景：
① 分析训练过程中算子可能存在的溢出、梯度爆炸与消失等问题；
② 获取图层的输出数据，分析特征图的信息；
③ 网络迁移场景下的对比分析；

案例一：
中间文件IR说明

MindSpore

1、IR计算图

```python
class Net(nn.Cell):
    def __init__(self):
        super().__init__()
        self.add = ops.Add()
        self.sub = ops.Sub()
        self.mul = ops.Mul()
        self.div = ops.Div()

    def func(x, y):
        return self.div(x, y)

    def construct(self, x, y):
        a = self.sub(x, 1)
        b = self.add(a, y)
        c = self.mul(b, self.func(a, b))
        return c
```

```
#IR entry      : @1_construct_wrapper.21
#attrs         :
#Total params  : 2

%para1_x : <Tensor[Float32]x()>
%para2_y : <Tensor[Float32]x()>

#Total subgraph : 3

subgraph attr:
Undeterminate : 0
subgraph @2_construct.22(%para3_x, %para4_y) {
  %0(a) = Sub(%para3_x, Tensor(shape=[], dtype=Float32, value= 1)) {instance name: sub} primitive_attrs: {input_names: [x, y],
      : (<Tensor[Float32]x()>, <Tensor[Float32]x()>) -> (<Tensor[Float32]x()>)
      # In file train.py(34)/        a = self.sub(x, 1)/
  %1(b) = Add(%0, %para4_y) {instance name: add} primitive_attrs: {input_names: [x, y], output_names: [output]}
      : (<Tensor[Float32]x()>, <Tensor[Float32]x()>) -> (<Tensor[Float32]x()>)
      # In file train.py(35)/        b = self.add(a, y)/
  %2([CNode]5) = call @3_func.23(%0, %1)
      : (<Tensor[Float32]x()>, <Tensor[Float32]x()>) -> (<Tensor[Float32]x()>)
      # In file train.py(36)/        c = self.mul(b, self.func(a, b))/
  %3(c) = Mul(%1, %2) {instance name: mul} primitive_attrs: {input_names: [x, y], output_names: [output]}
      : (<Tensor[Float32]x()>, <Tensor[Float32]x()>) -> (<Tensor[Float32]x()>)
      # In file train.py(36)/        c = self.mul(b, self.func(a, b))/
  Return(%3)
      : (<Tensor[Float32]x()>)
      # In file train.py(37)/        return c/
}
```

入口图

输入参数

子图

关联用户代码

```
subgraph attr:
subgraph @1_construct_wrapper.21() {
  %0([CNode]2) = call @2_construct.22(%para1_x, %para2_y)
      : (<Tensor[Float32]x()>, <Tensor[Float32]x()>) -> (<Tensor[Float32]x()>)
      # In file train.py(37)/        return c/
  Return(%0)
      : (<Tensor[Float32]x()>)
      # In file train.py(37)/        return c/
}
```

案例二：
自定义函数属性报错

MindSpore

1、自定义参数类，定义设置参数方法

```
class LayerParams:
    def __init__(self, dtype: str):
        self._type = dtype

    def get_weights(self, shape):
        nn_param = initializer(XavierUniform(), shape, mindspore.float32)
        nn_param = mindspore.Parameter(nn_param)
        return nn_param

class MyCell(nn.Cell):
    def __init__(self):
        super().__init__()
        self._fc_params = LayerParams("fc")
        self.matmul = ops.MatMul()

    def _fc(self, inputs, output_size):
        width = inputs.shape[-1]
        weight = self._fc_params.get_weights((width, output_size))
        return weight

    def construct(self, x, output_size):
        weight = self._fc(x, output_size)
        output = self.matmul(x, weight)
        return output
```

2、报错信息

```
Traceback (most recent call last):
  File "test_compiler_cls_customization.py", line 54, in <module>
    test_cls_customization()
  File "test_compiler_cls_customization.py", line 50, in test_cls_customization
    outputs = net(inputs, 5)
  File "lib/python3.7/site-packages/mindspore/nn/cell.py", line 572, in __call__
    out = self.compile_and_run(*args)
  File "lib/python3.7/site-packages/mindspore/nn/cell.py", line 951, in compile_and_run
    self.compile(*inputs)
  File "lib/python3.7/site-packages/mindspore/nn/cell.py", line 924, in compile
    _cell_graph_executor.compile(self, *inputs, phase=self.phase, auto_parallel_mode=self._auto_parallel_mode)
  File "lib/python3.7/site-packages/mindspore/common/api.py", line 1087, in compile
    result = self._graph_executor.compile(obj, args_list, phase, self._use_vm_mo
TypeError: Not supported to get attribute for InterpretedObject: '<__main__.Laye                    0xffff3aa16e50>'
The first argument should be a NameSpace, but got AbstractScalar(Type: External,
Value: InterpretedObject: '<__main__.LayerParams object at 0xffff3aa16e50>', Shape: NoShape)
The function call stack (See file 'rank_0/om/analyze_fail.dat' for more details.
Get instructions about `analyze_fail.dat` at https://www.mindspore.cn/search?inputValue=analyze_fail.dat):
# 0 In file test_compiler_cls_customization.py(41)
        weight = self._fc(x, output_size)
                 ^
# 1 In file test_compiler_cls_customization.py(37)
        weight = self._fc_params.get_weights((width, output_size))
```

Python调用栈

报错描述

报错代码行

3、调试分析：

① 获取报错描述，初步判断是静态图语法解析报错场景，涉及功能不支持。
② 构建一个简化样例，复现报错。
③ 使用pynative模式验证，问题不复现，可确认是静态图的语法解析问题。
④ 查看官网静态图语法支持介绍，网络构造组件中不包含函数属性方法。

静态图语法说明

网络构造组件

类别	内容
`Cell` 实例	mindspore/nn/*、自定义Cell。
`Cell` 实例的成员函数	Cell的construct中可以调用其他类成员函数。
`dataclass` 实例	使用@dataclass装饰的类。
`Primitive` 算子	mindspore/ops/operations/*
`Composite` 算子	mindspore/ops/composite/*
`constexpr` 生成算子	使用@constexpr生成的值计算算子。
函数	自定义Python函数、前文中列举的系统函数。

4、报错原因：静态图语法中，网络构造组件不支持自定义函数的属性方法。新版本中已规划。

案例三：
自定义的数据增强方法报错

MindSpore

1、自定义增强函数实现裁剪图片四个角和中间，**输入图像为PIL**

```
def corp_4_corner_1_center(img, size=224, is_resize=False):


    if isinstance(size, int):
        size = (size, size)
    elif isinstance(size, (tuple, list)) and len(size) == 2:
        size = size
    else:
        raise TypeError("Size should be a single number or a list/tuple (h, w) of length 2.")

    # PIL image.size returns in (width, height) order
    img_width, img_height = img.size
    crop_height, crop_width = size
    crop_top = int(round((img_height - crop_height) / 2.))
    crop_left = int(round((img_width - crop_width) / 2.))
    if is_resize:
        img_15_channal = np.concatenate([image.resize((180, 180), 2) for image in img_list], axis=2)
    else:
        img_15_channal = np.concatenate([image for image in img_list], axis=2)
    return img_15_channal
```

2、py_transform调用自定义增强函数

```
transforms_list = [
    #py_vision.ToPIL(),
    py_vision.Decode(),
    py_vision.RandomHorizontalFlip(0.5),
    py_vision.Resize((256, 256)),
    py_vision.RandomColorAdjust(0.4, 0.4, 0.4, 0.1),
    lambda image: corp_4_corner_1_center(img=image,size=224,is_resize=True),
    py_vision.ToTensor(),

    # 4D stack of 5 images
    ]
transforms_img_list = [
    lambda image: numpy.reshape(image, (batch_size*5, 3, 180, 180)),
]
```

3、**报错内容**

```
[ERROR] MD(189221,python):2021-08-08-11:02:25.285.074 [mindspore/ccsrc/minddata/dataset/util/task.cc:67] operator()] Thread ID 281471729455584 Exception thrown from PyFunc. map operation: [PyFunc] failed. The corresponding data files: /data/workspace/mindspore_dataset/imagenet/imagenet_original/train/n02099267/n02099267_2889.JPEG. Error description:
TypeError: Invalid object with type `<class 'PIL.Image.Image'>` and value `<PIL.Image.Image image mode=RGB size=180x180 at 0xFFFF6132EA58>`.
```

4、调试分析：

① 获取报错描述和python调用栈，是MD模块即数据加载&数据处理部分报错。
② 构建一个数据处理样例，**复现报错**。
③ **报错复现**，np.concatenate()函数执行报错。
④ 查询np.concatenate接口说明可知，输入参数要求是**numpy.ndarray**

```
At:
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/_extends/parse/parser.py(252): get_obj_type
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/ops/primitive.py(556): _run_op
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/common/api.py(75): wrapper
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/ops/primitive.py(188): __call__
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/numpy/array_ops.py(427): concatenate
  dataset.py(73): corp_4_corner_1_center
  dataset.py(119): <lambda>
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/dataset/transforms/py_transforms_util.py(47): compose
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/dataset/transforms/py_transforms.py(134): __call__
  /root/archiconda3/envs/ci3.7/lib/python3.7/site-packages/mindspore/dataset/transforms/validators.py(259):
```

报错原因：np.concatenate()函数的输入参数为PIL.Image对象，与要求不符。

https://bbs.huaweicloud.com/forum/thread-184123-1-1.html

案例四：
问题知识图谱工具
TroubleShooter

Error Message ----> TroubleShooter ----> Failure Analysis Report

MindSpore

1 使用方法

```
import troubleshooter as ts

context.set_context(mode=mindspore.PYNATIVE_MODE,

class Net(nn.Cell):
    def __init__(self):
        super().__init__()
        self.add = ops.Add()
        self.sub = ops.Sub()
        self.mul = ops.Mul()
        self.div = ops.Div()
    def func(x, y):
        return self.div(x, y)
    def construct(self, x, y):
        a = self.sub(x, 1)
        b = self.add(a, y)
        c = self.mul(b, self.func(a, a, b))
        return c

@ts.proposal()
def main():
    input1 = Tensor(3, mstype.float32)
    input2 = Tensor(2, mstype.float32)
    net = Net()
    out = net(input1, input2)
    print(out)
```

2 使用前

```
[EXCEPTION] ANALYZER(31946,7f6f03941740,python):2021-09-18-15:10:49.094.863 [mindspore/ccsrc/pipeline/jit/static_analy
FunctionGraph ID : func.18
NodeInfo: In file test.py(19)
    def func(x, y):

Traceback (most recent call last):
  File "test.py", line 31, in <module>
    out = net(input1, input2)
  File "/home/workspace/mindspore/mindspore/nn/cell.py", line 404, in __call__
    out = self.compile_and_run(*inputs)
  File "/home/workspace/mindspore/mindspore/nn/cell.py", line 682, in compile_and_run
    self.compile(*inputs)
  File "/home/workspace/mindspore/mindspore/nn/cell.py", line 669, in compile
    _cell_graph_executor.compile(self, *inputs, phase=self.phase, auto_parallel_mode=self._auto_parallel_mode)
  File "/home/workspace/mindspore/mindspore/common/api.py", line 542, in compile
    result = self._graph_executor.compile(obj, args_list, phase, use_vm, self.queue_name)
TypeError: mindspore/ccsrc/pipeline/jit/static_analysis/stack_frame.cc:85 DoJump] The parameters number of the functio
FunctionGraph ID : func.18
NodeInfo: In file test.py(19)
    def func(x, y):

The function call stack (See file '/home/workspace/mindspore/rank_0/om/analyze_fail.dat' for more details):
# 0 In file test.py(26)
        return c
        ^
# 1 In file test.py(25)
        c = self.mul(b, self.func(a, a, b))
                        ^
```

3 使用后

```
                                MindSpore FAR(Failure Analysis Report)
+---------------+-----------------------------------------------------------------------------------------------------+
|      项目     | 描述                                                                                                |
+---------------+-----------------------------------------------------------------------------------------------------+
|   版本信息:   | r1.7                                                                                                |
|   执行模式:   | PyNative Mode                                                                                       |
|   配置设备:   | CPU                                                                                                 |
|   可能原因:   | 自定义函数参数定义列表，与函数输入参数列表不匹配引起报错                                                            |
| 示例错误代码: | +-------------------------------------------+                                                       |
|               | > class Net(nn.Cell):                                                                               |
|               | >     ...                                                                                           |
|               | >     def func(x, y):                                                                               |
|               | >         return self.div(x, y)                                                                     |
|               | >     def construct(self, x, y):                                                                    |
|               | >         a=self.sub(x, 1)                                                                          |
|               | >         b=self.add(a, y)                                                                          |
|               | >         c=self.mul(b,self.func(a,a,b))                                                            |
|               | >                         ^~~~~~~~~参数与函数定义不匹配                                                     |
|               | >         return c                                                                                  |
|               | +-------------------------------------------+                                                       |
|   处理建议:   | 参考函数定义，修改自定义函数调用参数.                                                                          |
| 示例正确代码: | +-------------------------------------------+                                                       |
|               | > class Net(nn.Cell):                                                                               |
|               | >     ...                                                                                           |
|               | >     def func(x, y):                                                                               |
|               | >         return self.div(x, y)                                                                     |
|               | >     def construct(self, x, y):                                                                    |
|               | >         a=self.sub(x, 1)                                                                          |
|               | >         b=self.add(a, y)                                                                          |
|               | >         c=self.mul(b, self.func(a,b))                                                             |
|               | >                         ^~~~~~~~~修改参数与函数定义匹配                                                   |
|               | >         return c                                                                                  |
|               | +-------------------------------------------+                                                       |
|   相关案例:   | 1.使用analyze_fail.dat分析函数参数错误案例:                                                                  |
|               | https://www.mindspore.cn/docs/programming_guide/zh-CN/r1.6/read_ir_files.html#analyze-fail-dat       |
+---------------+-----------------------------------------------------------------------------------------------------+
```

附录
功能调试资料汇总

MindSpore

动态图模式应用：https://www.mindspore.cn/tutorials/zh-CN/r1.7/advanced/pynative_graph/pynative.html

查看中间文件（IR)： https://www.mindspore.cn/tutorials/experts/zh-CN/r1.7/debug/mindir.html

混合精度：https://www.mindspore.cn/tutorials/experts/zh-CN/r1.7/others/mixed_precision.html

Dump功能： https://www.mindspore.cn/tutorials/experts/zh-CN/r1.7/debug/dump.html

官网FAQ： https://www.mindspore.cn/docs/zh-CN/r1.7/faq/installation.html

华为云论坛： https://bbs.huaweicloud.com/forum/forum-1076-2704-1.html

开源社区： https://gitee.com/mindspore/mindspore/issues

THANK YOU