2.2 onnx转om

对于开源框架的网络模型（如Caffe、TensorFlow等），不能直接在昇腾AI处理器上运行推理，需要先使用ATC（Ascend Tensor Compiler）工具将开源框架的网络模型转换为适配昇腾AI处理器的离线模型（*.om文件）

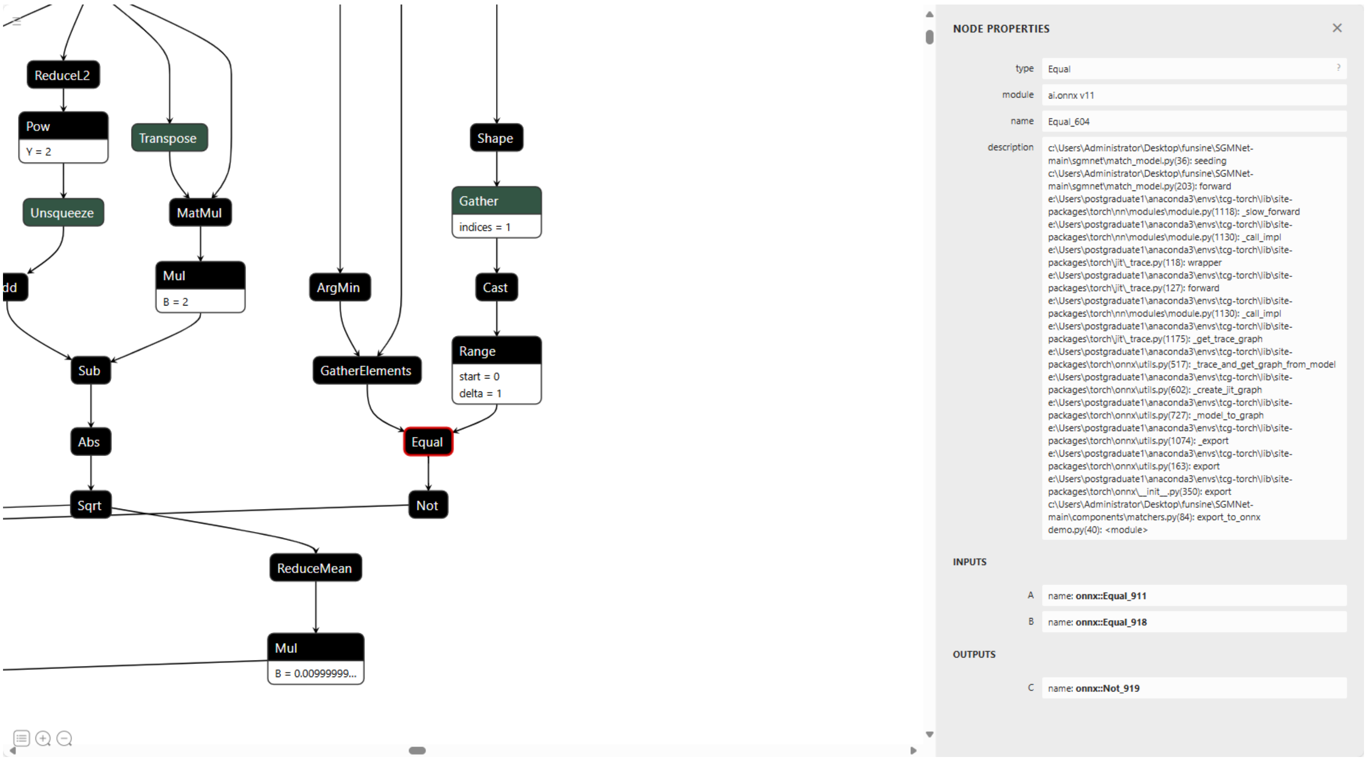
```
atc --model=/home/tcg/VisualHabitFusion/model.onnx --framework=5 --output=/home/tcg/VisualHabitFusion/matcher --soc_version=Ascend310B4 --input_shape="x1_in:1,4000,3;x2_in:1,4000,3;desc1_in:1,4000,256;desc2_in:1,4000,256"
```

问题1：`The Equal_604 op dtype is not same, type1:DT_INT32, type2:DT_INT64`

```
ATC run failed, Please check the detail log, Try 'atc --help' for more information
E10042: GenerateOfflineModel execute failed.
    TraceBack (most recent call last):
    op[Equal_604], The Equal_604 op dtype is not same, type1:DT_INT32, type2:DT_INT64[FUNC:CheckTwoInputDtypeSame][FILE:util.cc][LINE:116]
    Verifying Equal_604 failed.[FUNC:InferShapeAndType][FILE:infershape_pass.cc][LINE:137]
    Call InferShapeAndType for node:Equal_604(Equal) failed[FUNC:Infer][FILE:infershape_pass.cc][LINE:119]
    process pass InferShapePass on node:Equal_604 failed, ret:4294967295[FUNC:RunPassesOnNode][FILE:base_pass.cc][LINE:571]
    build graph failed, graph id:0, ret:1343242270[FUNC:BuildModelWithGraphId][FILE:ge_generator.cc][LINE:1615]
    GenerateOfflineModel execute failed.
```

问题定位：

1.在netron中将onnx模型的结构打印出来，寻找问题节点的位置



NODE PROPERTIES

type	Equal
module	ai.onnx v11
name	Equal_604
description	c:\Users\Administrator\Desktop\funsine\SGMNet-main\sgmnet\match_model.py(36): seeding c:\Users\Administrator\Desktop\funsine\SGMNet-main\sgmnet\match_model.py(203): forward e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\nn\modules\module.py(1118): _slow_forward e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\nn\modules\module.py(1130): _call_impl e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\jit_trace.py(118): wrapper e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\jit_trace.py(127): forward e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\nn\modules\module.py(1130): _call_impl e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\jit_trace.py(1175): _get_trace_graph e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\utils.py(517): _trace_and_get_graph_from_model e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\utils.py(602): _create_jit_graph e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\utils.py(727): _model_to_graph e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\utils.py(1074): _export e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\utils.py(163): export e:\Users\postgraduate1\anaconda3\envs\tcg-torch\lib\site-packages\torch\onnx\__init__.py(350): export c:\Users\Administrator\Desktop\funsine\SGMNet-main\components\matchers.py(84): export_to_onnx demo.py(40): <module>

INPUTS

A	name: **onnx::Equal_911**
B	name: **onnx::Equal_918**

OUTPUTS

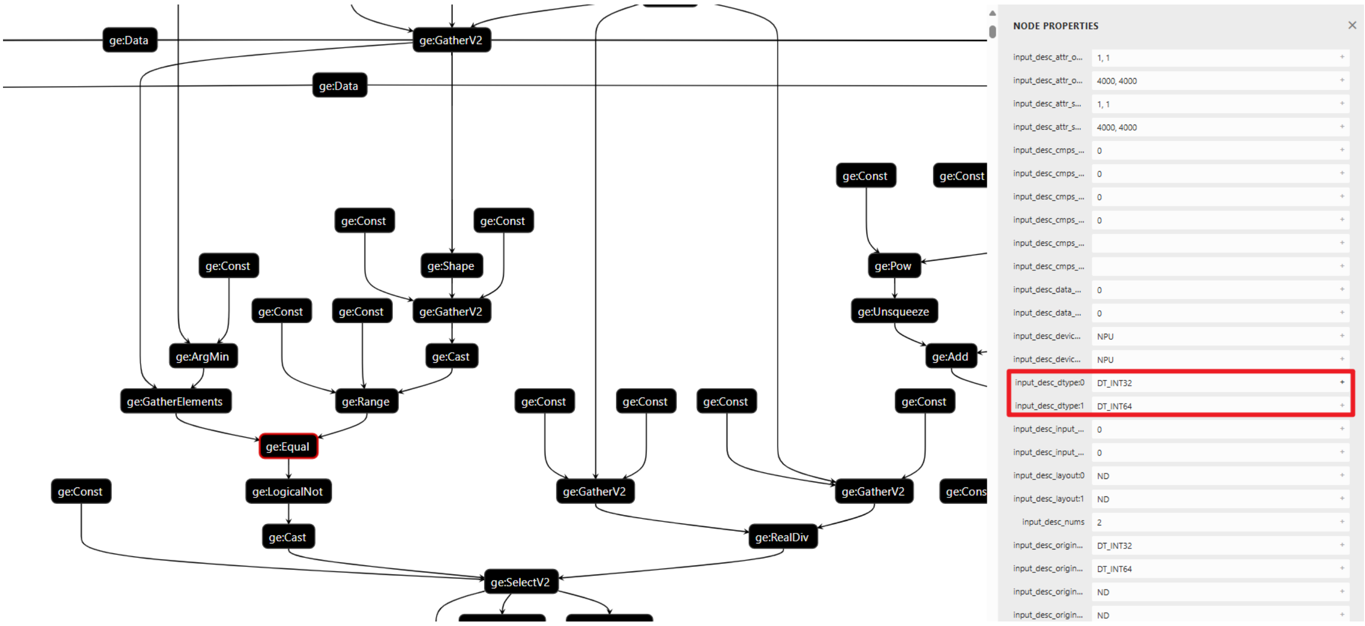
C	name: **onnx::Not_919**

可以定位到源码为：

```
if use_mc:
    
mask_not_mutual=nn_index2.gather(dim=-1,index=nn_index1)!=torch.arange(nn_index1.s
hape[1],device='cuda')
match_score[mask_not_mutual]=-1
```

打印`!=`两侧的数据类型为`int64,int64`,在pytorch中类型是正确的，推测是atc工具的问题。

2.追寻atc问题： 使用 `export DUMP_GE_GRAPH=2`生成Dump图，在Dump图目录下找到`ge_onnx_***_graph_0_after_infershape.pbtxt`，可在GE图中确定Equal算子确实存在两个input，算子的dtype类型分别为int32和int64。



NODE PROPERTIES

input_desc_attr_o...	1, 1
input_desc_attr_o...	4000, 4000
input_desc_attr_s...	1, 1
input_desc_attr_s...	4000, 4000
input_desc_cmps_...	0
input_desc_cmps_...	0
input_desc_cmps_...	0
input_desc_cmps_...	0
input_desc_cmps_...	
input_desc_cmps_...	
input_desc_data_...	0
input_desc_data_...	0
input_desc_devic...	NPU
input_desc_devic...	NPU
input_desc_dtype:0	DT_INT32
input_desc_dtype:1	DT_INT64
input_desc_input_...	0
input_desc_input_...	0
input_desc_layout:0	ND
input_desc_layout:1	ND
input_desc_nums	2
input_desc_origin...	DT_INT32
input_desc_origin...	DT_INT64
input_desc_origin...	ND
input_desc_origin...	ND

解决：在equal前添加cast算子，将int32转为int64

```
import onnx
from onnx import helper, TensorProto
# 加载现有的 ONNX 模型
onnx_model = onnx.load('C:\\Users\\Administrator\\Desktop\\funsine\\SGMNet-main\\demo\\model.onnx')
# 获取模型中的计算图
graph = onnx_model.graph
# 查找目标节点 GatherElements_596 和 Equal_604
gather_node_name = "GatherElements_596"
equal_node_name = "Equal_604"
gather_output = None
# 先找到 GatherElements_596 节点的输出
for node in graph.node:
    if node.name == gather_node_name:
        gather_output = node.output[0]
        break
# 确保找到了 GatherElements_596 的输出
if gather_output is None:
    raise ValueError(f"Node {gather_node_name} not found in the graph.")
# 创建一个 Cast 节点，将 GatherElements_596 的输出转换为 int64
cast_output = gather_output + "_casted"
cast_node = helper.make_node(
    'Cast',  # 算子类型
    inputs=[gather_output],  # Cast的输入是GatherElements的输出
    outputs=[cast_output],  # Cast的输出
    to=TensorProto.INT64  # 将输出转换为 int64
)
# 将 Cast 节点添加到计算图中
graph.node.append(cast_node)
# 更新 Equal_604 节点，将它的输入改为 Cast 节点的输出
for node in graph.node:
    if node.name == equal_node_name:
        for i, input_name in enumerate(node.input):
            if input_name == gather_output:
                node.input[i] = cast_output  # 修改 Equal_604 的输入
                break
# 保存修改后的模型
onnx.save(onnx_model, 'model_with_cast.onnx')
print("Cast node successfully inserted between GatherElements_596 and Equal_604.")
```

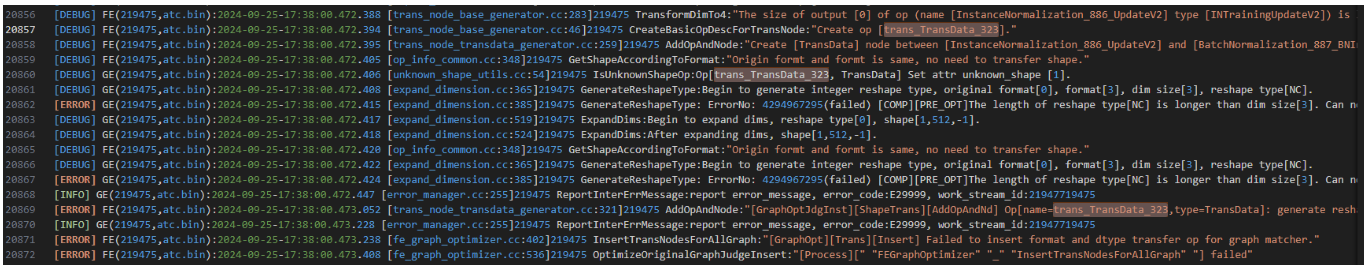
问题2：Op[name=trans_TransData_323,type=TransData]: generate reshape type mask of input failed

```
ATC run failed, Please check the detail log, Try 'atc --help' for more information
E10042: GenerateOfflineModel execute failed.
    TraceBack (most recent call last):
```

```
    [GraphOptJdgInst][ShapeTrans][AddOpAndNd]
Op[name=trans_TransData_323,type=TransData]: generate reshape type mask of input
failed.[FUNC:AddOpAndNode][FILE:trans_node_transdata_generator.cc][LINE:321]
    [GraphOpt][Trans][Insert] Failed to insert format and dtype transfer op for
graph matcher.[FUNC:InsertTransNodesForAllGraph][FILE:fe_graph_optimizer.cc]
[LINE:402]
    Call OptimizeOriginalGraphJudgeInsert failed, ret:-1,
engine_name:AIcoreEngine,
graph_name:matcher[FUNC:OptimizeOriginalGraphJudgeInsert][FILE:graph_optimize.cc]
[LINE:251]
    build graph failed, graph id:0, ret:-1[FUNC:BuildModelWithGraphId]
[FILE:ge_generator.cc][LINE:1615]
    GenerateOfflineModel execute failed.
```

问题定位：

在atc命令后添加`--log=debug`打印日志，进入日志文件查找`trans_TransData_323`如下：



```
[DEBUG] FE(219475,atc.bin):2024-09-25-17:38:00.472.388 [trans_node_base_generator.cc:283]219475 TransformDimTo4:"The size of output [0] of op (name [InstanceNormalization_886_UpdateV2] type [INTrainingUpdateV2]) is
[DEBUG] FE(219475,atc.bin):2024-09-25-17:38:00.472.394 [trans_node_base_generator.cc:46]219475 CreateBasicOpDescForTransNode:"Create op [trans_TransData_323]."
[DEBUG] FE(219475,atc.bin):2024-09-25-17:38:00.472.395 [trans_node_transdata_generator.cc:259]219475 AddOpAndNode:"Create [TransData] node between [InstanceNormalization_886_UpdateV2] and [BatchNormalization_887_BNI
[DEBUG] FE(219475,atc.bin):2024-09-25-17:38:00.472.405 [op_info_common.cc:348]219475 GetShapeAccordingToFormat:"Origin formt and formt is same, no need to transfer shape."
[DEBUG] GE(219475,atc.bin):2024-09-25-17:38:00.472.406 [unknown_shape_utils.cc:54]219475 IsUnknownShapeOp:Op[trans_TransData_323, TransData] Set attr unknown_shape [1].
[DEBUG] GE(219475,atc.bin):2024-09-25-17:38:00.472.408 [expand_dimension.cc:365]219475 GenerateReshapeType:Begin to generate integer reshape type, original format[0], format[3], dim size[3], reshape type[NC].
[ERROR] GE(219475,atc.bin):2024-09-25-17:38:00.472.415 [expand_dimension.cc:385]219475 GenerateReshapeType: ErrorNo: 4294967295(failed) [COMP][PRE_OPT]The length of reshape type[NC] is longer than dim size[3]. Can n
[DEBUG] GE(219475,atc.bin):2024-09-25-17:38:00.472.417 [expand_dimension.cc:519]219475 ExpandDims:Begin to expand dims, reshape type[0], shape[1,512,-1].
[DEBUG] GE(219475,atc.bin):2024-09-25-17:38:00.472.418 [expand_dimension.cc:524]219475 ExpandDims:After expanding dims, shape[1,512,-1].
[DEBUG] FE(219475,atc.bin):2024-09-25-17:38:00.472.420 [op_info_common.cc:348]219475 GetShapeAccordingToFormat:"Origin formt and formt is same, no need to transfer shape."
[DEBUG] GE(219475,atc.bin):2024-09-25-17:38:00.472.422 [expand_dimension.cc:365]219475 GenerateReshapeType:Begin to generate integer reshape type, original format[0], format[3], dim size[3], reshape type[NC].
[ERROR] GE(219475,atc.bin):2024-09-25-17:38:00.472.424 [expand_dimension.cc:385]219475 GenerateReshapeType: ErrorNo: 4294967295(failed) [COMP][PRE_OPT]The length of reshape type[NC] is longer than dim size[3]. Can n
[INFO] GE(219475,atc.bin):2024-09-25-17:38:00.472.447 [error_manager.cc:255]219475 ReportInterErrMessage:report error_message, error_code:E29999, work_stream_id:21947719475
[ERROR] FE(219475,atc.bin):2024-09-25-17:38:00.473.052 [trans_node_transdata_generator.cc:321]219475 AddOpAndNode:"[GraphOptJdgInst][ShapeTrans][AddOpAndNd] Op[name=trans_TransData_323,type=TransData]: generate resh
[INFO] GE(219475,atc.bin):2024-09-25-17:38:00.473.228 [error_manager.cc:255]219475 ReportInterErrMessage:report error_message, error_code:E29999, work_stream_id:21947719475
[ERROR] FE(219475,atc.bin):2024-09-25-17:38:00.473.238 [fe_graph_optimizer.cc:402]219475 InsertTransNodesForAllGraph:"[GraphOpt][Trans][Insert] Failed to insert format and dtype transfer op for graph matcher."
[ERROR] FE(219475,atc.bin):2024-09-25-17:38:00.473.408 [fe_graph_optimizer.cc:536]219475 OptimizeOriginalGraphJudgeInsert:"[Process][" "FEGraphOptimizer" "_" "InsertTransNodesForAllGraph" "] failed"
```

分析上面的日志流程如下：

```
1.节点创建:
[DEBUG] Create op [trans_TransData_323]: 成功创建了一个新的操作节点
trans_TransData_323。
[DEBUG] Create [TransData] node between [InstanceNormalization_886_UpdateV2] and
[BatchNormalization_887_BNInferenceD] success!: 该节点在
InstanceNormalization_886_UpdateV2 和 BatchNormalization_887_BNInferenceD 之间创建
成功。

2.形状处理:
[DEBUG] GetShapeAccordingToFormat:"Origin formt and formt is same, no need to
transfer shape.": 输入的形状格式与原始格式相同，因此无需进行形状转换。
[DEBUG] IsUnknownShapeOp:Op[trans_TransData_323, TransData] Set attr unknown_shape
[1]: 标记此操作的形状为未知。

3.生成重塑类型:
[DEBUG] GenerateReshapeType:Begin to generate integer reshape type...: 开始生成整数
重塑类型，原始格式和目标格式信息被记录下来。
[ERROR] GenerateReshapeType: ErrorNo: 4294967295(failed)... The length of reshape
type[NC] is longer than dim size[3].: 生成重塑类型时出错，错误信息表明重塑类型 NC 的长
度大于维度大小 3，无法生成整数重塑类型。

4. 扩展维度:
[DEBUG] ExpandDims:Begin to expand dims...: 开始扩展维度操作。
```

```
[DEBUG] ExpandDims:After expanding dims, shape[1,512,-1].: 扩展维度后的形状信息。

5.再度生成重塑类型:
由于扩展操作后再次尝试生成重塑类型，但再次出现相同的错误信息，表明在此过程中仍然无法满足重
塑要求。

错误报告:
[ERROR] AddOpAndNode:"... generate reshape type mask of input failed.": 在尝试添加
操作和节点时，生成输入的重塑类型掩码失败，导致整个操作无法完成。
```

总结：atc尝试在`InstanceNormalization_886_UpdateV2`和`BatchNormalization_887_BNInferenceD`插入一个数据转换算子 `trans_TransData_323`,但是在图计算中执行节点操作时遇到的形状问题，主要是由于重塑类型与实际维度不匹配，导致无法完成操作。核心报错原因是上图 `[error]`这一行:

```
[ERROR] GE(219475,atc.bin):2024-09-25-17:38:00.472.415
[expand_dimension.cc:385]219475 GenerateReshapeType: ErrorNo: 4294967295(failed)
[COMP][PRE_OPT]The length of reshape type[NC] is longer than dim size[3]. Can not
generate integer reshape type
```

问题解决：