Below figure shows the current process of export and deployment. The first method shows how to deploy etlt model to DS4.0 or <u>https://github.com/NVIDIA-AI-IOT/deepstream\_4.x\_apps</u>. The second method shows how to generate TRT engine and deploy it to DS4.0 or <u>https://github.com/NVIDIA-AI-IOT/deepstream\_4.x\_apps</u>.

Data Type	Network Type	Export Process	Deploy etlt model	Deploy TRT engine
int8	Classification/ Detectnet_V2	training <u>tit-int8-tensorfile</u> <u>cal.tensor</u> <u>tit-export</u> etit model and <u>cal.bin</u>	cal.bin	etit model tit-converter exgene DS4.0
	Faster_rcnn	tt file tit-export and cal.bin	etit model _4.x_apps	etit model tit-converter, engine cal.bin
	SSD	SSD does not support int8 export	NA	NA
fp16/fp32	Classification/ Detectnet_V2	tit file tit-export etit model	etit model -> DS4.0	etit model tit-converter engine DS4.0
	Faster_rcnn/ SSD	tit file tit-export etit model	etit model deepstream	etit model til:converters engine deepstream _4.x_apps

2. Below figure shows a new deploy method for TRT engine. Note: One TRT engine is only runnable on NVGPU.



Please refer to preprocess/postprocess code which is exposed in C++ in

- nvdsinfer\_customparser\_frcnn\_uff folder from <u>https://github.com/NVIDIA-AI-IOT/deepstream\_4.x\_apps</u>
- nvdsinfer\_customparser\_ssd\_uff folder from <u>https://github.com/NVIDIA-AI-IOT/deepstream\_4.x\_apps</u>
- deepstream\_sdk\_v4.0.1\_jetson/sources/libs/nvdsinfer\_customparser/nv dsinfer\_custombboxparser.cpp from <u>https://developer.nvidia.com/deepstream-download</u>