CLARA TRAIN SDK RELEASE NOTES

RN-09303-002_v2.0 | March 2019

Release Notes

TABLE OF CONTENTS

Chapter 1. CLARA TRAIN SDK: TRANSFER LEARNING

Description

Clara Train SDK: Transfer Learning is a containerized Python package that simplifies deep learning tasks on medical image data (such as segmentation from 3D CT/MRI) and enables user to train or fine-tune pretrained models and export them for TensorRT based inference. This is the Early Access release of this product.

Key Features

Clara Train SDK: Transfer Learning includes:

- Configurable framework to simplify deep learning tasks from medical images, such 3D segmentation from CT/MRI images or 2D X-ray classification.
- Optimized pre-trained deep learning models and reference applications for Medical Imaging use cases.
- Model adaptation and retraining that is easy to use in heterogeneous multiple GPU environments.
- Model Export API for easier deployment of applications to TensorRT based inference.

NVIDIA Clara Train SDK: Transfer Learning includes:

- An annotation server - a python based server that is used when the docker run command is executed.
- Examples - Example code demonstrating how to integrate the SDK into third party applications showing API functionality.
- Utilities - Helper classes to assist in the integration efforts on the client side for third party applications.

C++ client side plugin for MITK is a separate component by itself that is delivered using gitlab. The client side component will have two plugins.

- 3D segmentation plugin to get segmentation mask.

- 2D polygon editing to enable smart polygon editing.

Contents

Components included in this release:

- Transfer Learning and AI Assisted Annotation docker container
- Jupyter notebook with sample workflows
- Pre-trained models
- Getting Started Guide containing usage and installation instructions

Software Requirements

- Ubuntu 16.04 LTS
- NVIDIA GPU driver v410.xx or above
- nvidia-docker 2.0 installed, instructions: https://github.com/NVIDIA/nvidia-docker.

Hardware Requirements

Recommended

- 1 GPU or more
- 16 GB GPU memory
- 8 core CPU
- 32 GB system RAM
- 80 GB free disk space

Known Issues

- When training or fine tuning the models in multi-GPU setting on small number of training data, it is recommended to adjust the learning rate provided in the configuration files, e.g. multiple the learning rate by the GPU number as is recommended in https://arxiv.org/pdf/1706.02677.pdf.
- Dynamically optimizing TensorRT graphs, part of tlt-export when run with the ` —trt_dynamic_mode` flag, will fail when optimizing subgraphs with less than 4 TensorFlow nodes. Nvidia recommends setting the **`--trt_min_seg_size`** value to a value greater than 3 to avoid this issue if using custom tlt-export configuration.
- Classification models with TensorRT optimizations will cause tlt-infer and tlt-evaluate to fail due to an input ValueError. NVIDIA recommends using the frozen `.fzn.pb` version of classification models for tlt-infer and tlt-evaluate. Alternatively, re-export the model and experiment with larger **`--trt_min_seg_size`** values.
- This software was tested on DGX 1V.

Resolved Issues

There are no resolved issues in this release.

Chapter 2. CLARA TRAIN SDK: AI ASSISTED ANNOTATION

Description

Clara Train SDK: AI Assisted Annotation (AIAA) is an SDK enabling customers to bring AI-assisted workflows into medical imaging applications accelerating the annotation process with faster 3D segmentation. This is the Early Access release of this product.

Key Features

The release includes these features:

- AI Assisted Annotation for 3D segmentation and analysis.
- 13 deep learning models for organ annotations.
- Smart polygon editing for faster and efficient corrections.
- Flexible C++ and Python API to integrate into a medical imaging application.
- Adapt and consistently increase annotation accuracy over time using NVIDIA Clara Train SDK: Transfer Learning workflow.

Contents

Components included in this release:

- AIAA docker on NVIDIA GPU Cloud including Transfer Learning.
- Jupyter notebooks with sample workflows in the container.
- Open Source code on Github for client side integration.
- Getting Started Guide containing usage and installation instructions.

Software Requirements

- Ubuntu 16.04 LTS
- nvidia-docker 2.0 installed, instructions: https://github.com/NVIDIA/nvidia-docker
- NVIDIA GPU driver v410.xx or above

Hardware Requirements

Recommended

- 1 GPU or more
- 16 GB GPU memory
- 8 core CPU
- 32 GB system RAM
- 80 GB free disk space

Known Issues

- When training or fine tuning the models in multi-GPU setting on small number of training data, it is recommended to adjust the learning rate provided in the configuration files, e.g. multiple the learning rate by the GPU number as is recommended in https://arxiv.org/pdf/1706.02677.pdf.
- Dynamically optimizing TensorRT graphs, part of tlt-export when run with the ` —trt_dynamic_mode` flag, will fail when optimizing subgraphs with less than 4 TensorFlow nodes. NVIDIA recommends setting the **`--trt_min_seg_size`** value to a value greater than 3 to avoid this issue if using custom tlt-export configuration.
- This software was tested on DGX 1V.

Resolved Issues

There are no resolved issues in this release.

Notice

Trademarks

Copyright