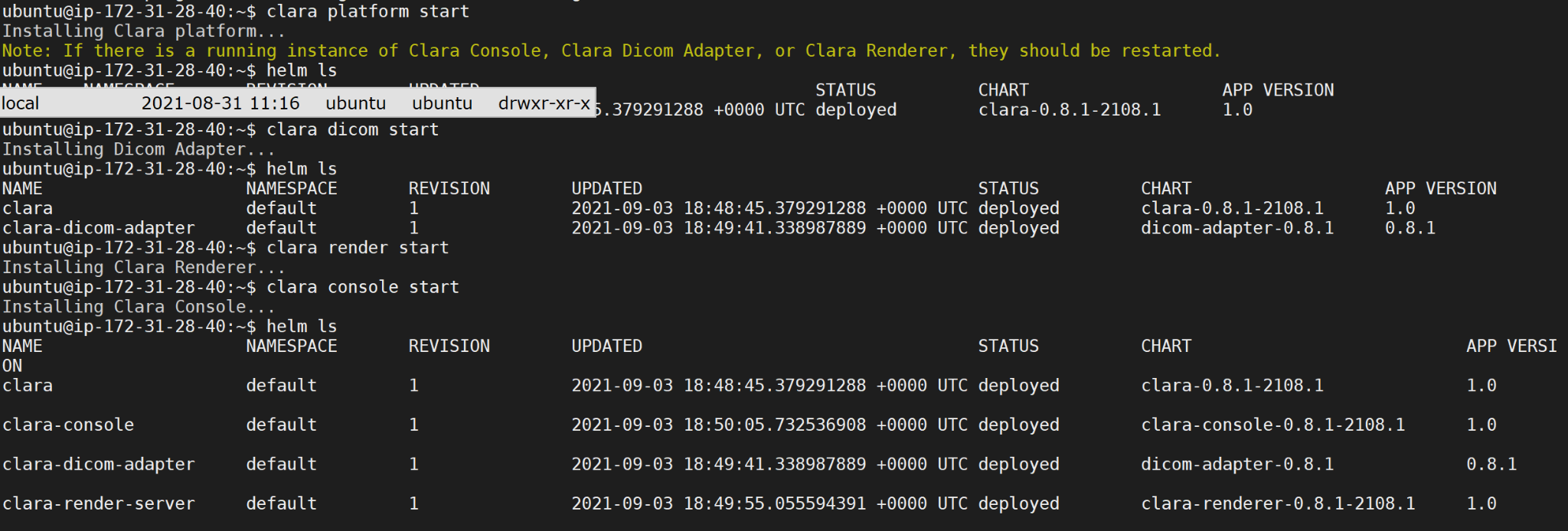
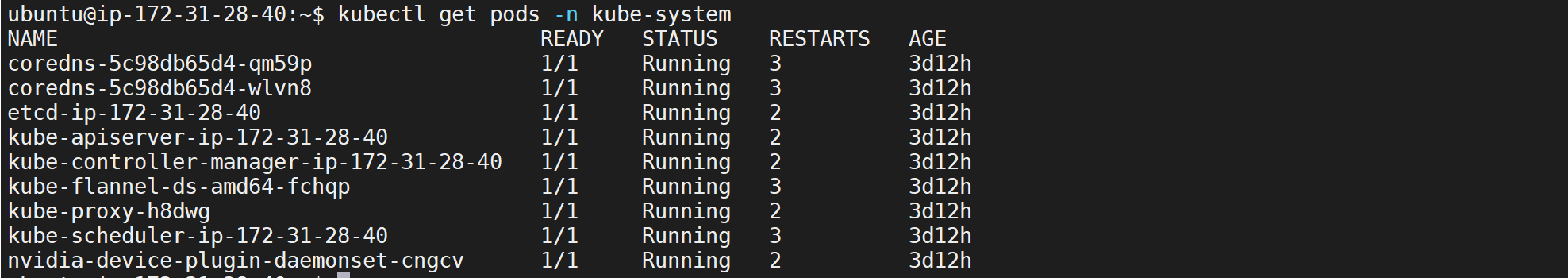
**Inference using AI Model**

Let's start clara back up again.

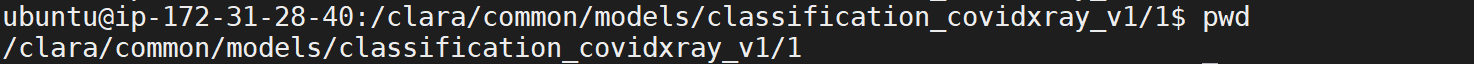


Deployed kubernetes pods



**Setting up TRITON inference server to host our model**

Create a folder structure as below



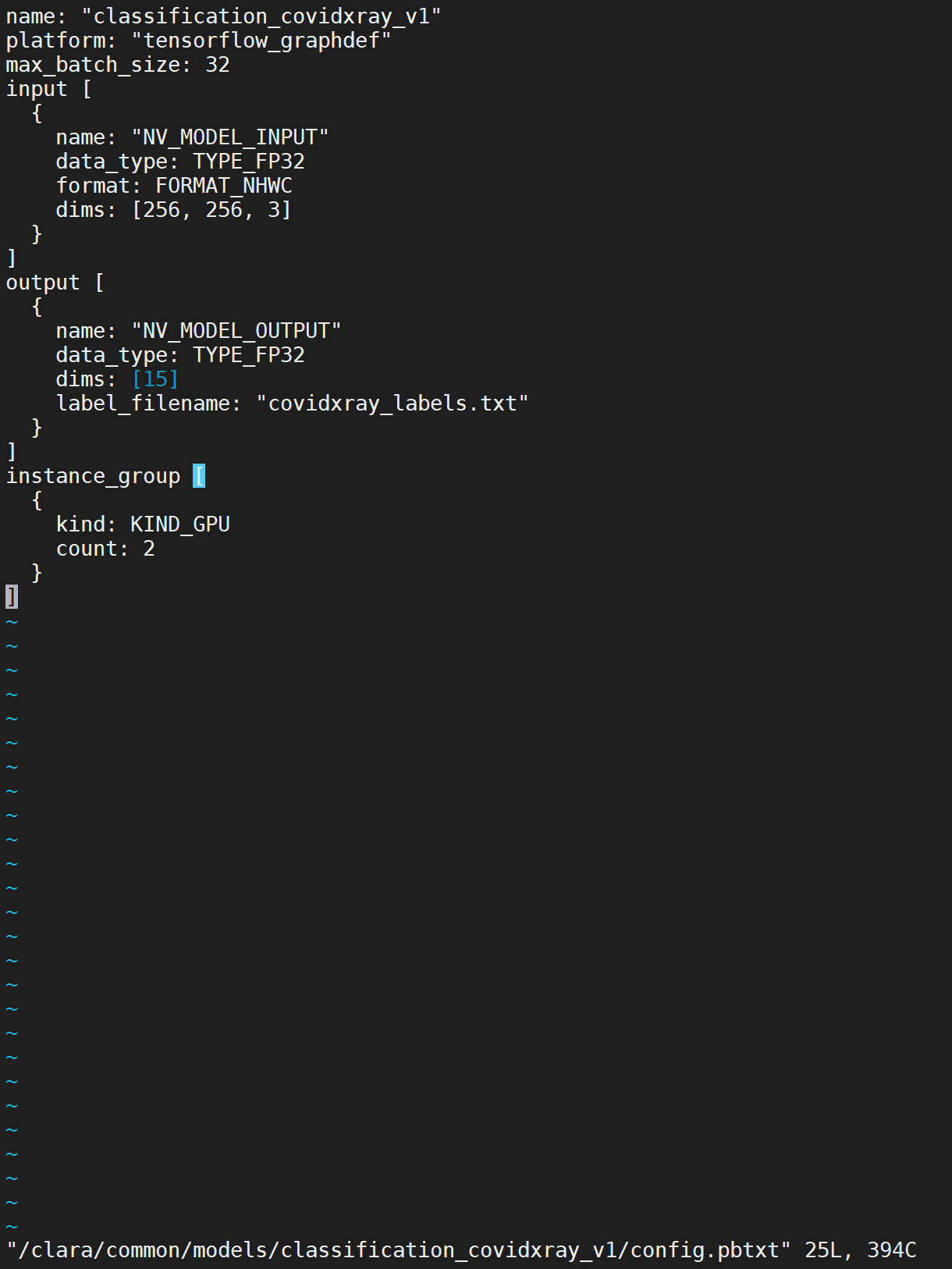
Move the refined model we created before to this directory.



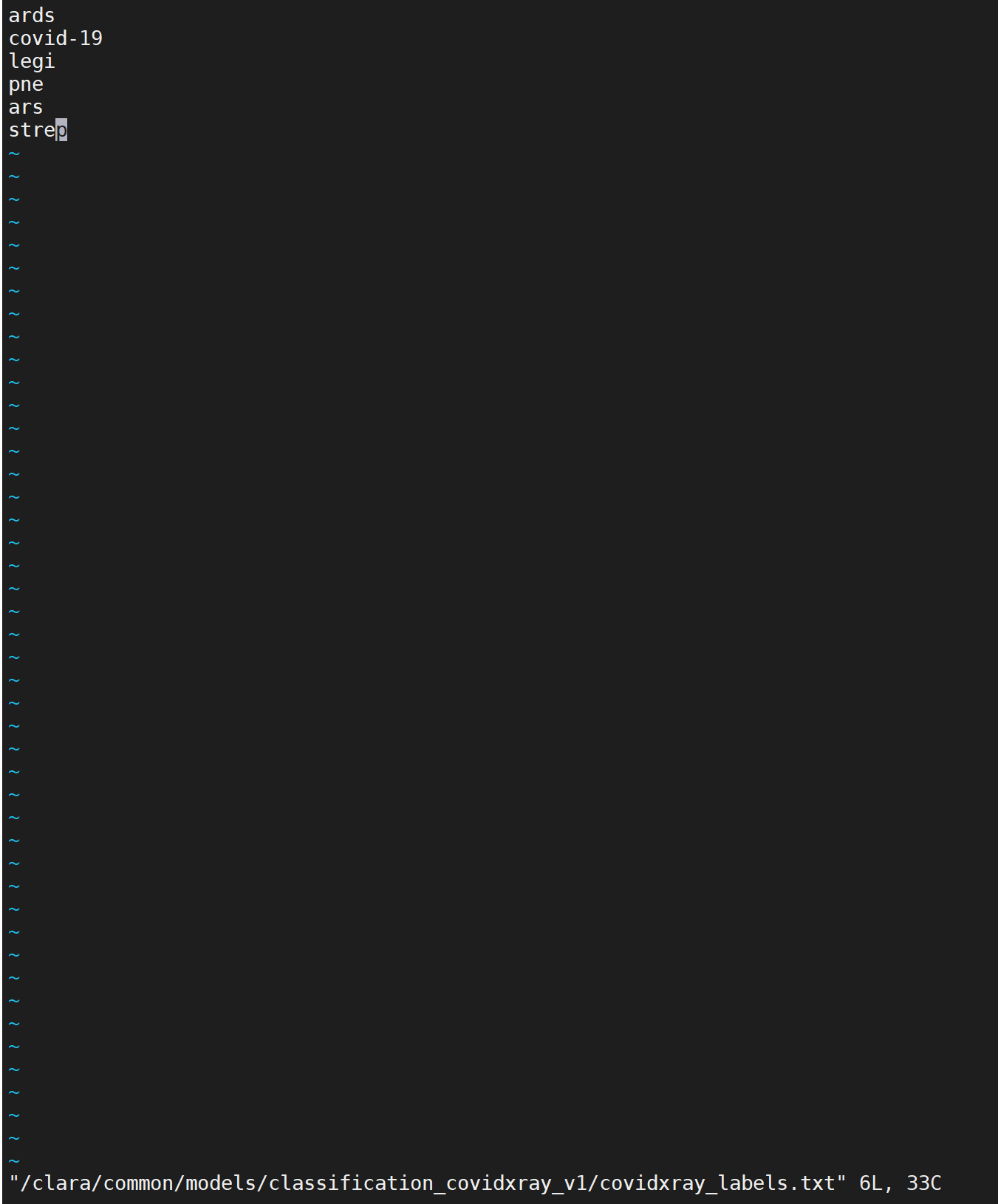
Refer: <https://blog.uplandr.com/2021/09/02/Fine-tune-a-Chest-Xray-Classification-Model-using-NVIDIA-Clara-Train.html>

Refer to this documentation to know about the directory structure: <https://docs.nvidia.com/deeplearning/triton-inference-server/master-user-guide/>

Create a file as below:



Create another file with our labels



**Create a Clara Deploy Operator**

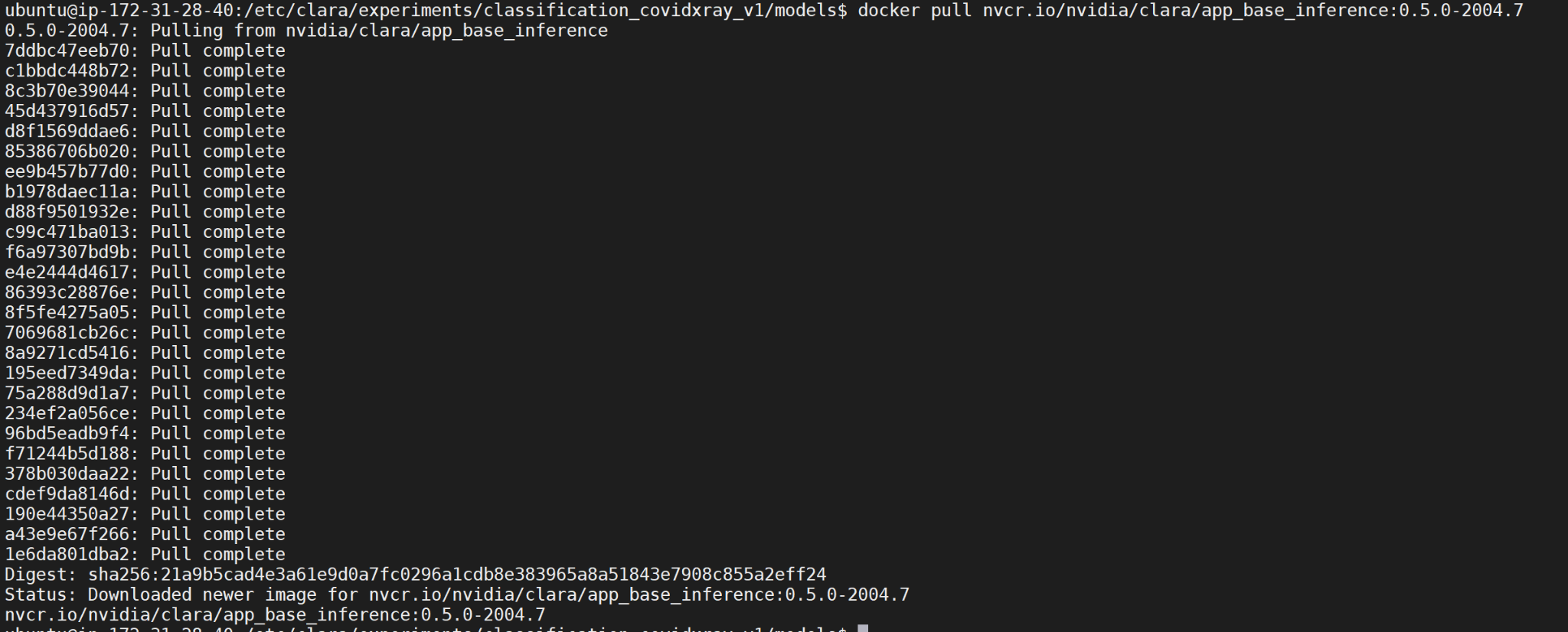
We will create a clara deploy operator. This operator will be running in a container independent of clara deploy and the operator can be made part of a deployment pipeline.

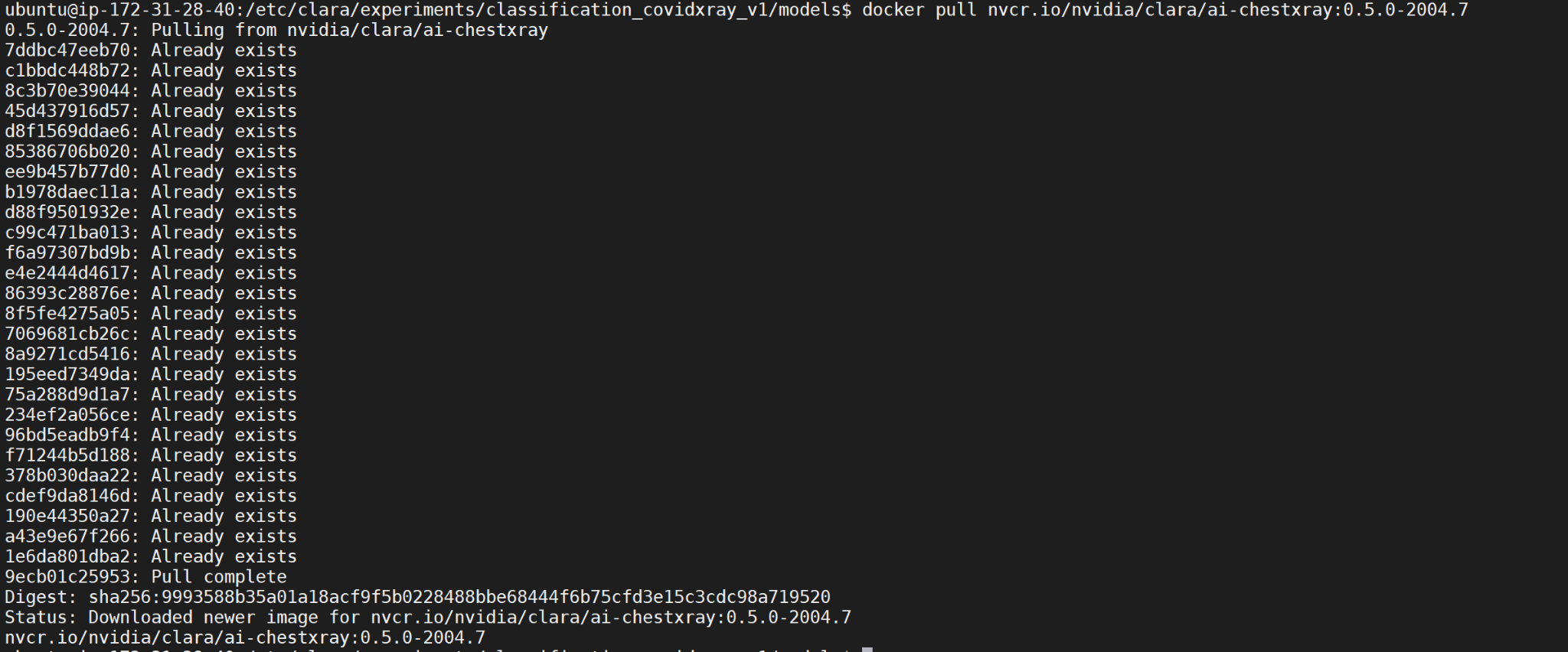
Steps are given in here - <https://ngc.nvidia.com/catalog/containers/nvidia:clara:app_base_inference>

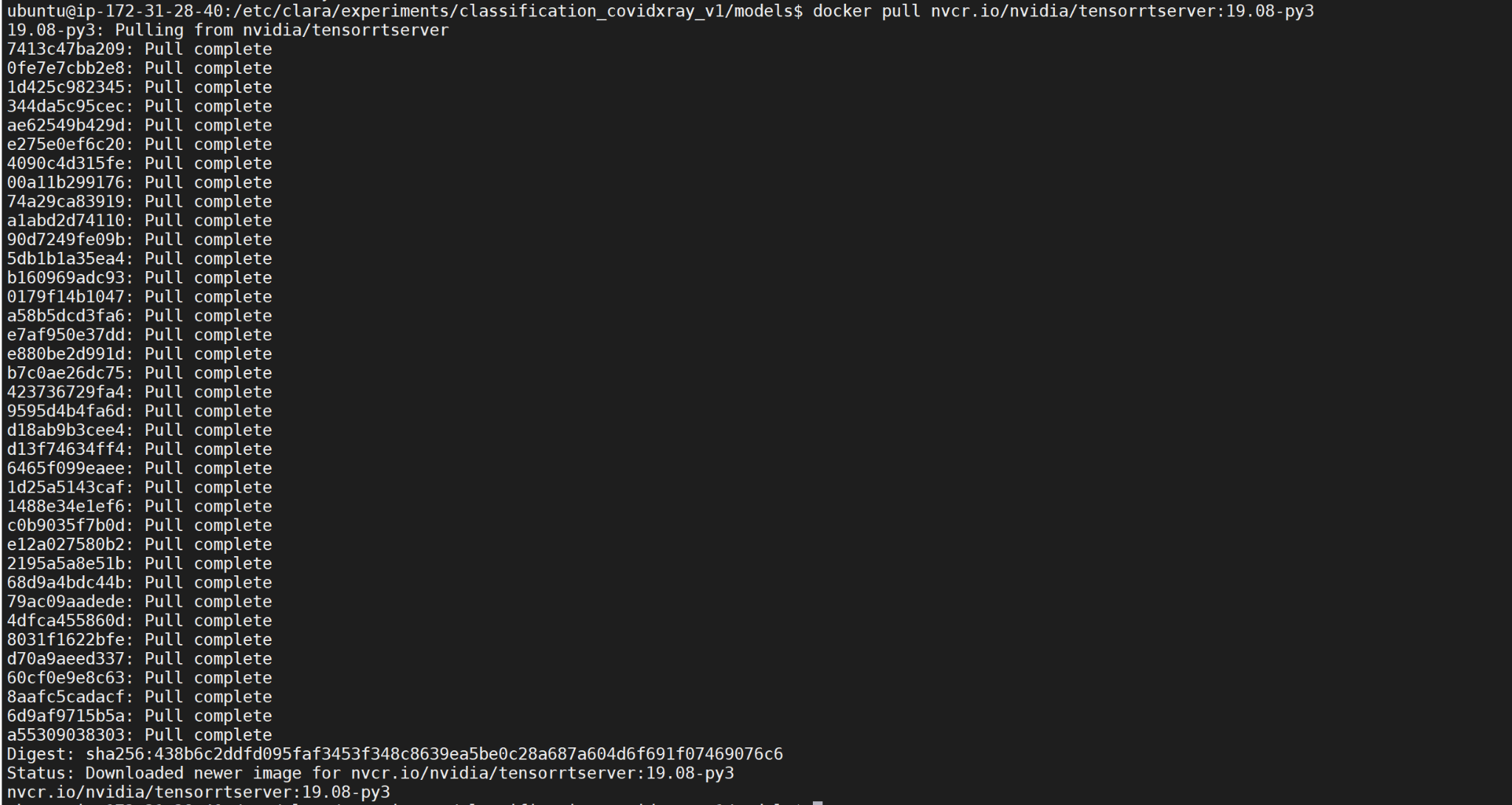
We need to grab these-

* Clara deploy base inference operator
* Clara chest classification operator
* TRITIS (Triton) container

Make sure you have your ngc connection or else rebuild connection to ngc with docker login nvcr.io



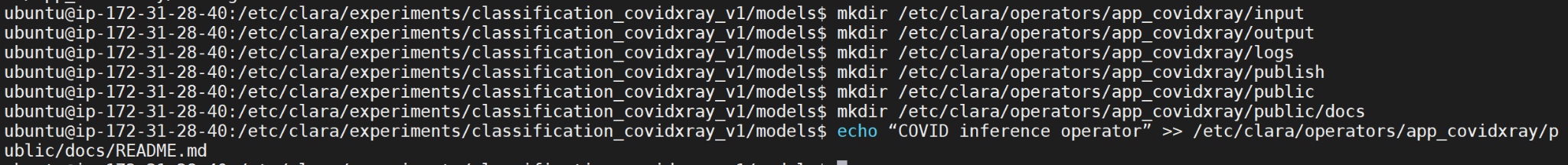




Retag the docker image as latest



Create a Operator directory structure



Run the chest xray operator docker container

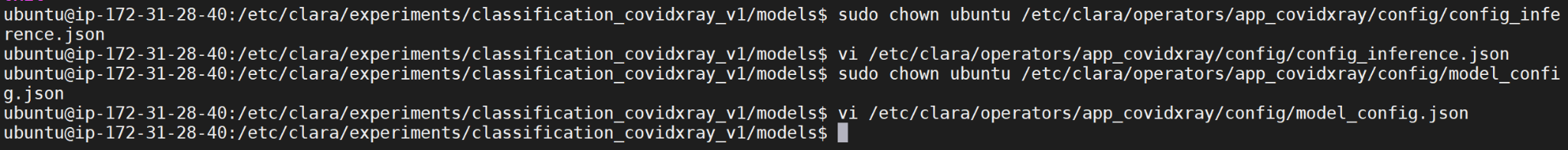


Copy 2 files from the container

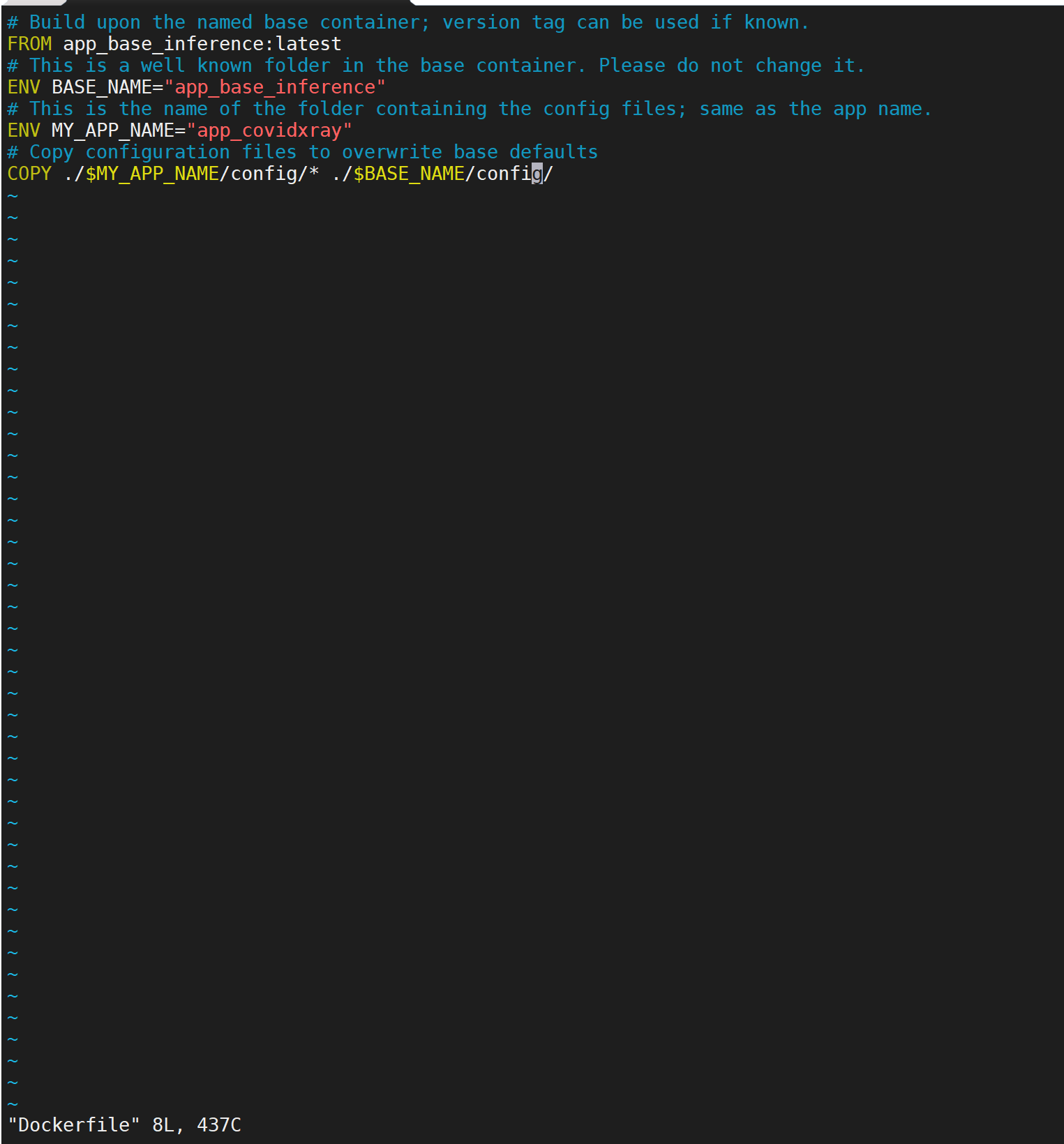




Exit from the container and change the owner for the files to your own. There are few changes to be made in these two files. Change the model to be used to “classification\_covidxray\_v1” from “classification\_cheastxray\_v1”. And in the config\_inference change the `subtrahend` and `divisor` to 128.



Create a Dockerfile with base as app\_base\_inference and copy the config files taken from the chestxray



**Test the custom operator**

We will run the operator outside of clara deploy pipeline using docker and a script.

Copy the script from the “executing with docker” section of the link - <https://ngc.nvidia.com/catalog/containers/nvidia:clara:app_base_inference>

Change the script as follows to make it suitable for our purpose.

Create a file

vi /etc/clara/operators/run\_covid\_docker.sh

Open the file run\_covid\_docker.sh and paste the script from “executing with docker” section of the link - <https://ngc.nvidia.com/catalog/containers/nvidia:clara:app_base_inference>

Need to make following edits:

Replace APP\_NAME with “app\_covidxray”

Replace MODEL\_NAME with “classification\_covidxray\_v1”.

The line that starts with nvidia-docker — replace $(pwd) with clara/common (so this part reads -v /clara/common/models/${MODEL\_NAME}:/models/${MODEL\_NAME}

In the line “-v $(pwd)/input:/input \”, replace $(pwd) with “/etc/clara/operators/app\_covidxray”

In the line “-v $(pwd)/output:/output \”, replace $(pwd) with “/etc/clara/operators/app\_covidxray”

In the line “-v $(pwd)/logs:/logs \”, replace $(pwd) with “/etc/clara/operators/app\_covidxray”

In the line “-v $(pwd)/publish:/publish \”, replace $(pwd) with “/etc/clara/operators/app\_covidxray”

Comment the lines as indicated in notes of the file if using NGC containers for testing.

Save and exit from the file.

Copy one image in our test input folder.

cp /etc/clara/experiments/covid-training-set/training-images/1-s2.0-S0929664620300449-gr2\_lrg-b.png /etc/clara/operators/app\_covidxray/input

Change permission of the script file and run the script

chmod 700 /etc/clara/operators/run\_covid\_docker.sh

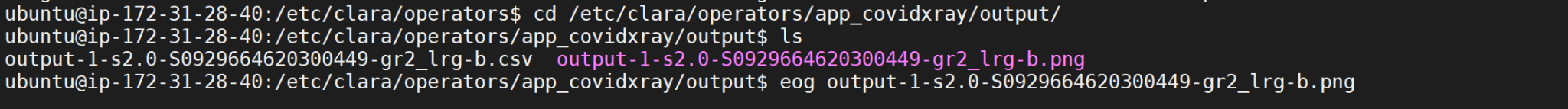
cd /etc/clara/operators/



To check the job was successful, check the output folder for a file with the inference



Check the output folder and display the image with labels and categories and % of chance

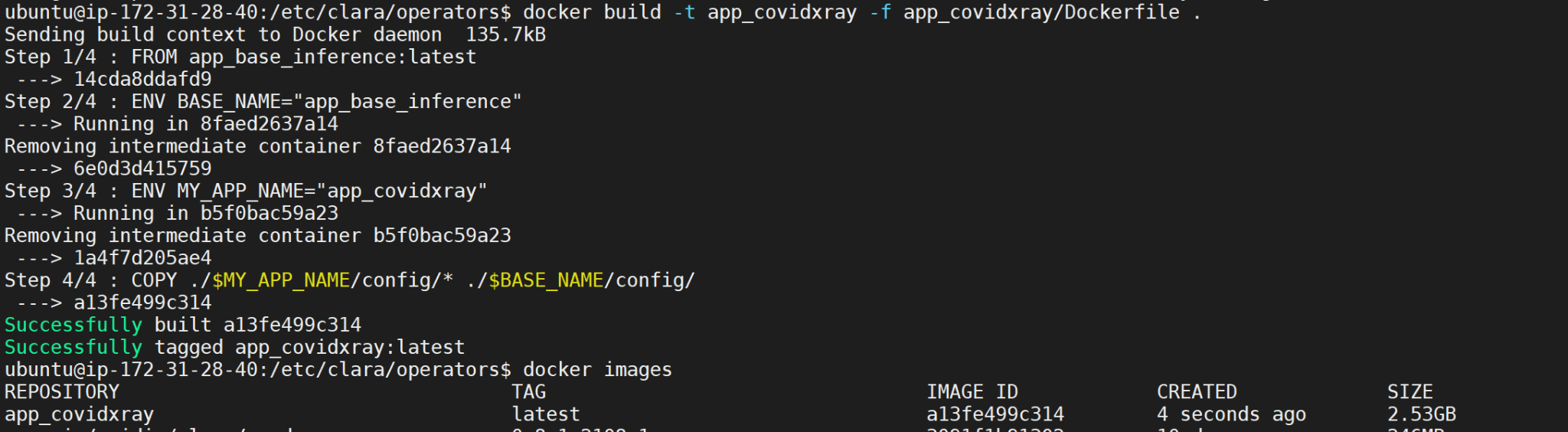


Output with inference shown in the picture!



**Create a Clara Deploy Pipeline for inference**

Create a clean docker build using the Dockerfile



The steps are described here - <https://docs.nvidia.com/clara/deploy/sdk/Applications/Pipelines/ChestxrayPipeline/public/docs/README.html>

<https://ngc.nvidia.com/catalog/containers/nvidia:clara:app_base_inference>

Start with a chest xray classification pipeline and change it to fit covid xray pipeline





Make some changes covidxray-pipeline.yaml file to fit it for our purpose

Change the container image to - app\_covidxray, and tag to latest

Remove the pull secrets part

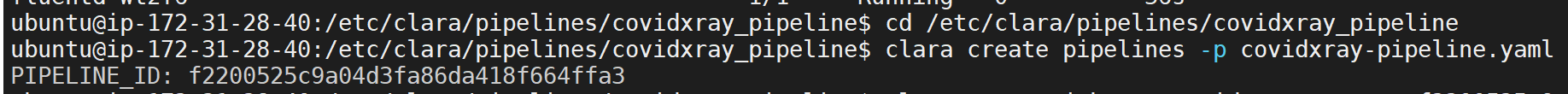
Change all reference of chest xray to covid xray

Note: Make sure the triton server version is appropriate. Pay attention to app\_base\_inference version, reference pipeline version (in this case clara\_ai\_chestxray\_pipeline) and triton server version. All these need to be in sync for the inference to work.

For the current example I am using app\_base\_inference ( not app\_base\_inference\_v2 ) and I used nvcr.io/nvidia/tensorrtserver tag 19.08-py3 (rather than tritonserver). Change the “Command” to “trtserver” if using tensorrtserver.

Save and exit

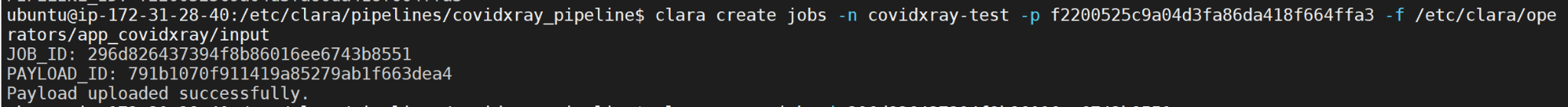
Now we are ready to create our covid xray pipeline



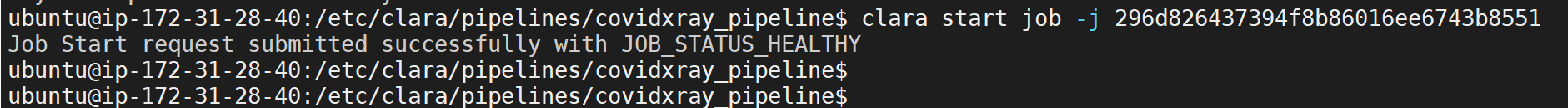
This will give you a pipeline id.

**Run test image through the pipeline**

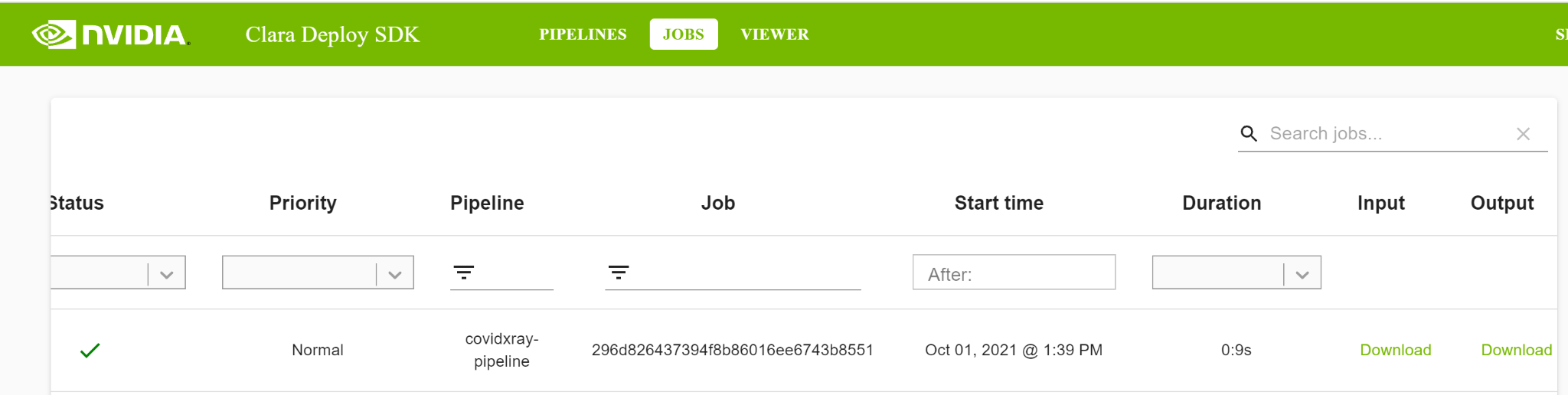
Now use the created pipeline to process one image from the input file.

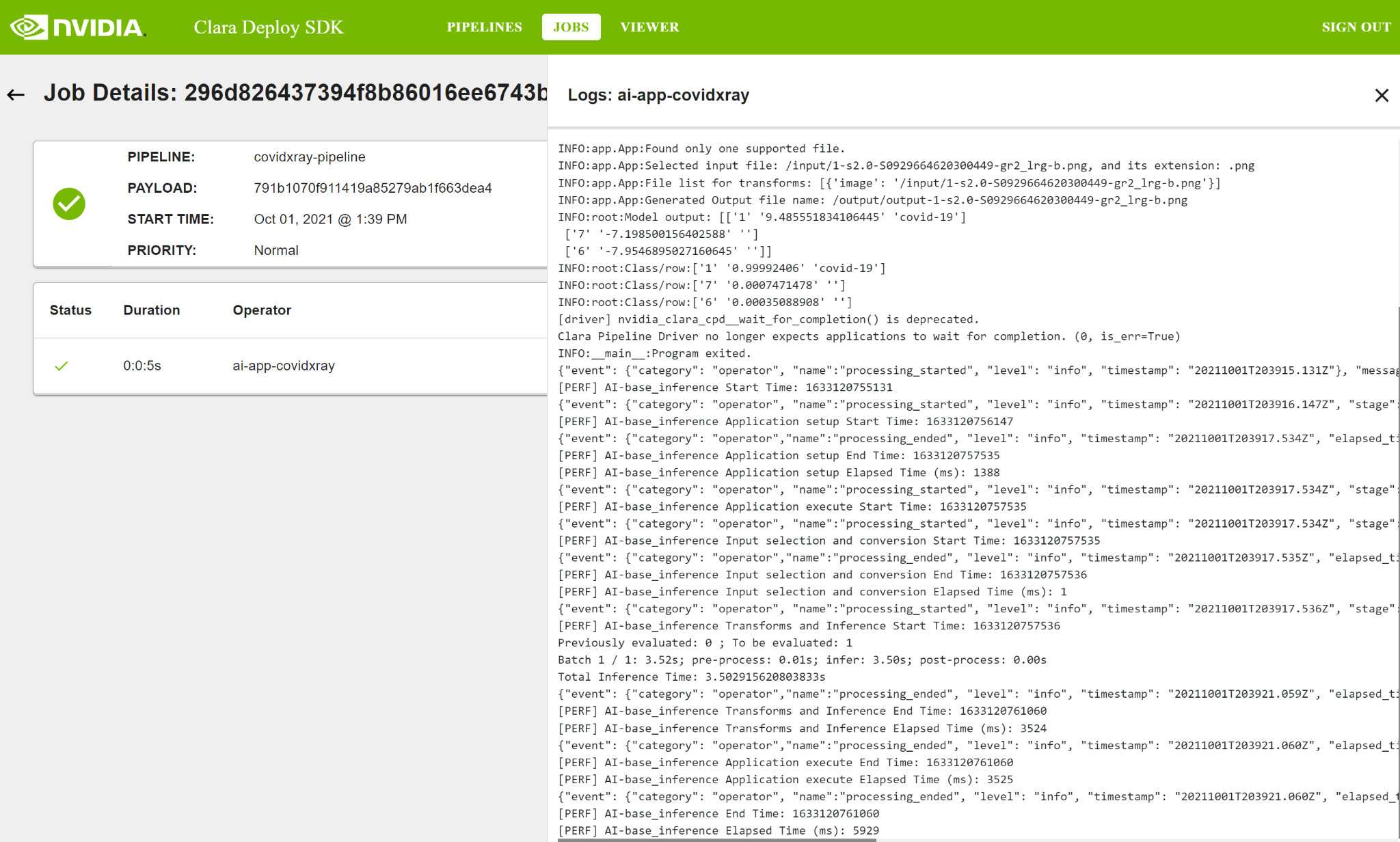


Manually start the job

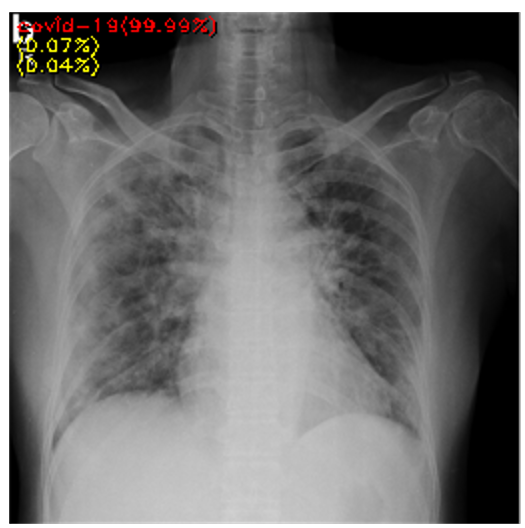


The completed pipeline view in Clara console (port 32002)





Output after download



Here you have it, your own model is used in inference through triton server and clara pipeline!