Submission Instructions

MOOD 2020

Table of contents

General Instructions

Introduction

This document contains instructions on how to **access the data**, how to create a **docker image** and how to **submit it** to the synapse portal to participate in our challenge.

Challenge details can be found at our web pages:
https://www.synapse.org/#!Synapse:syn21343101/wiki/599515

Technical Instructions

Data access

Training phase

1. Create an account at synapse and register for the challenge: https://www.synapse.org/#!Synapse:syn21343101/wiki/599515
2. Download the data as zip.

Test phase

Testing data won't be accessible during training. To be part of the challenge, you must **create a docker file and upload it to the synapse platform**. The challenge organizers will perform the complete evaluation. Please follow the instructions in the following sections to install, create and submit docker images.

Creation of docker images

Please install and use docker for submission: https://www.docker.com/get-started

You can build and use any docker base/ image you like.
There are already a good base docker images to build on for:

- Pytorch: https://hub.docker.com/r/pytorch/pytorch/
- Tensorflow https://www.tensorflow.org/install/docker

For GPU support you may need to install the NVIDIA Container Toolkit:
https://github.com/NVIDIA/nvidia-docker

For a simple example checkout out our Github: https://github.com/MIC-DKFZ/mood

Docker preparation

For the different tasks the docker needs the following scripts:

- Sample-level:
    - /workspace/run_sample_brain.sh input_folder output_folder (for the brain dataset)
    - /workspace/run_sample_abdom.sh input_folder output_folder (for the abdominal dataset)

- Pixel-level:
    - /workspace/run_pixel_brain.sh input_folder output_folder (for the brain dataset)
    - /workspace/run_pixel_abdom.sh input_folder output_folder (for the abdominal dataset)

The docker has to allow mounting the input folder to /mnt/data and the output folder to /mnt/pred. We will mount the input and output folder and pass them to the run scripts.

During testing, the docker image will be run with the following commands:

- Sample-level:

```
docker run --gpus all -v "<input folder>/:/mnt/data" -v
"<output folder>:/mnt/pred" docker-image-name
/workspace/run_sample_brain.sh /mnt/data /mnt/pred

docker run --gpus all -v "<input folder>/:/mnt/data" -v
"<output folder>:/mnt/pred" docker-image-name
/workspace/run_sample_abdom.sh /mnt/data /mnt/pred
```

- Pixel-level:

```
docker run --gpus all -v "<input folder>/:/mnt/data" -v
"<output folder>:/mnt/pred" docker-image-name
/workspace/run_pixel_brain.sh /mnt/data /mnt/pred

docker run --gpus all -v "<input folder>/:/mnt/data" -v
"<output folder>:/mnt/pred" docker-image-name
/workspace/run_pixel_abdom.sh /mnt/data /mnt/pred
```

In other words, it will **mount two folders**,
one at *"/mnt/data"* and
one at *"/mnt/pred"*.

Then, we will run the run scripts therefore, it must be written in such a way that it **will run your model automatically**.

The folder "*/mnt/data*" will contain the test data set, in the same way as the training data set, with the image dimensions equal to the image dimensions in the training data set.

Your model should process all scans and write the outputs to the given output directory.

Please be aware of the different output formats depending on your task:

- Sample-level: For each input file (e.g. input1.nii.gz) create a text file with the same name and an appended ".txt" file-ending (e.g. input1.nii.gz.txt) in the output directory and write out a single (float) score.
- Pixel-level: For each input file (e.g. input1.nii.gz) create a nifti file with the same dimensions and save it under the same (e.g. input1.nii.gz) in the output directory.

For more information have a look at our github example: https://github.com/MIC-DKFZ/mood

Docker submission

0. Becoming a certified Synapse user

Important: In order to use all docker functionality e.g. `docker push`, you must be a certified user. You can become a certified user by filling out the following quiz (*attention, the url contains a **":"** at the end!*): https://www.synapse.org/#!Quiz:

0. Test your docker on the Toy data

Please test your docker on the test data beforehand. See https://github.com/MIC-DKFZ/mood

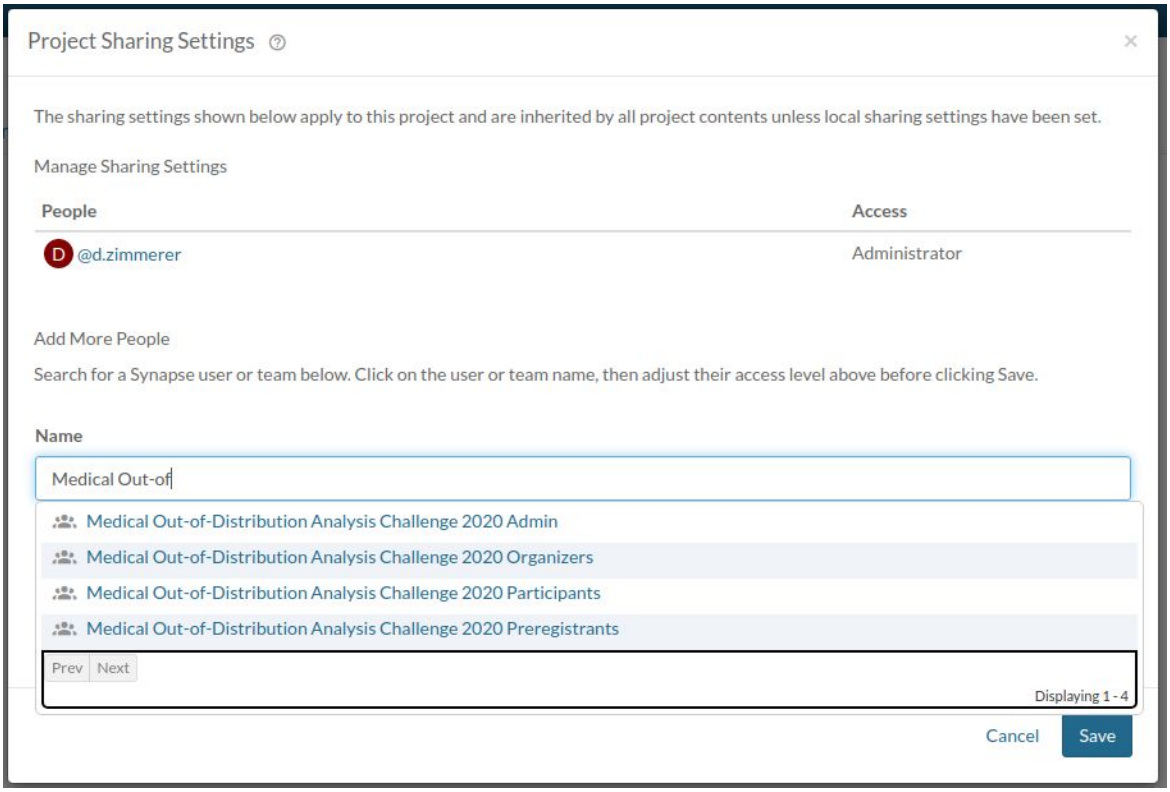
1. Create a new project on synapse

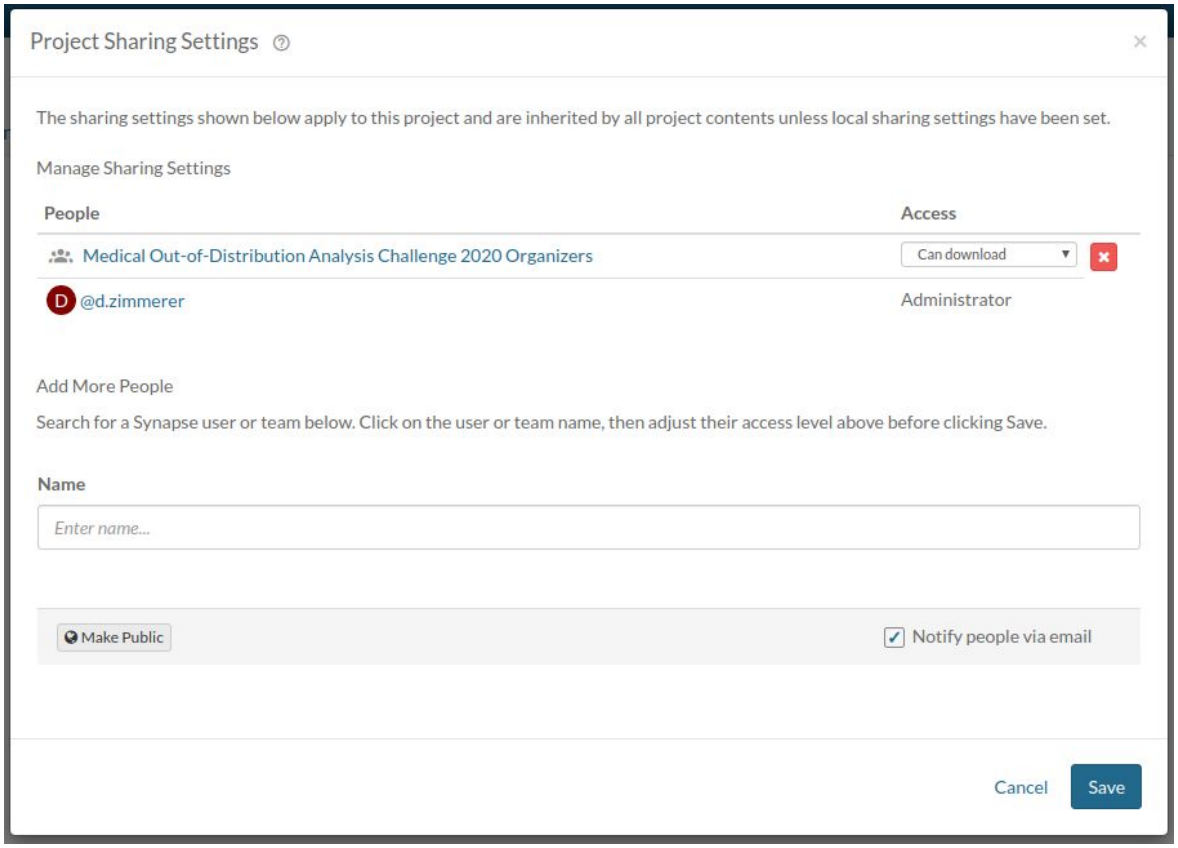
- To submit a docker file, you first need to **create a new project** on the synapse platform: (e.g:)

  MOOD_submission_<task>_<Your team name>

- **Note the Synapse Project ID** (e.g. *syn20482334*).

- Furthermore, once the project has been created, the organizing team of the challenge (***Medical Out of Distribution Analysis Challenge 2020 Organizers***) must be given download permissions to the project (Under *Project Settings -> Project Sharing Settings*), as shown in the following screenshots:


Project Sharing Settings
The sharing settings shown below apply to this project and are inherited by all project contents unless local sharing settings have been set.
Manage Sharing Settings
People
Access
@d.zimmerer
Administrator
Add More People
Search for a Synapse user or team below. Click on the user or team name, then adjust their access level above before clicking Save.
Name
Medical Out-of
Medical Out-of-Distribution Analysis Challenge 2020 Admin
Medical Out-of-Distribution Analysis Challenge 2020 Organizers
Medical Out-of-Distribution Analysis Challenge 2020 Participants
Medical Out-of-Distribution Analysis Challenge 2020 Preregistrants
Prev
Next
Displaying 1 - 4
Cancel
Save


Project Sharing Settings
The sharing settings shown below apply to this project and are inherited by all project contents unless local sharing settings have been set.
Manage Sharing Settings
People
Access
Medical Out-of-Distribution Analysis Challenge 2020 Organizers
Can download
@d.zimmerer
Administrator
Add More People
Search for a Synapse user or team below. Click on the user or team name, then adjust their access level above before clicking Save.
Name
Enter name...
Make Public
Notify people via email
Cancel
Save

2 . Upload docker-image to synapse

- `docker login docker.synapse.org` *(Enter synapse username and password)*

```
(base) rosst@mbi99:~$ sudo docker login docker.synapse.org
[sudo] password for rosst:
Username: schnobi1990
Password:
WARNING! Your password will be stored unencrypted in /home/rosst/.docker/config.
json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

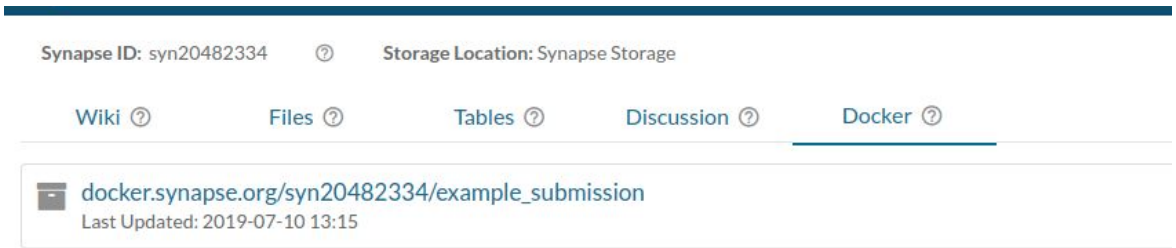
- `docker tag <imagename_local> docker.synapse.org/<ProjectSynapseID>/<imagename_synapse>`

```
rosst@mbi99:~$ sudo nvidia-docker tag pytorch_example_final_v2 docker.synapse.org/syn20482334/example_submission
rosst@mbi99:~$
```

- `Docker push docker.synapse.org/<SynapseProjectID>/<imagename>:latest`

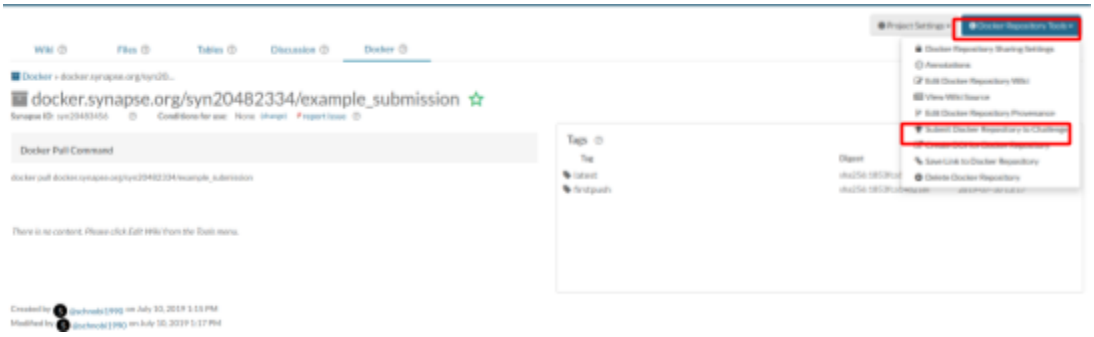
```
(base) rosst@mbi99:~$ sudo docker push docker.synapse.org/syn20482334/example_submission
[sudo] password for rosst:
The push refers to repository [docker.synapse.org/syn20482334/example_submission]
f12f1c6ace83: Preparing
bac58b2a5ace: Preparing
bac58b2a5ace: Layer already exists
92eb77d4bece: Layer already exists
3c3e897716e4: Layer already exists
838dfd1aef44: Layer already exists
80782a85330f: Layer already exists
46d1ee805a8f: Layer already exists
d6c0be28dc2d: Layer already exists
d798cbaaa513: Layer already exists
```

- Verify the success of your push
  You can verify on the Synapse website if the push was successful (If the upload fails, it might not be a **certified user**. Please refer to the upper paragraph “Becoming a certified user".)

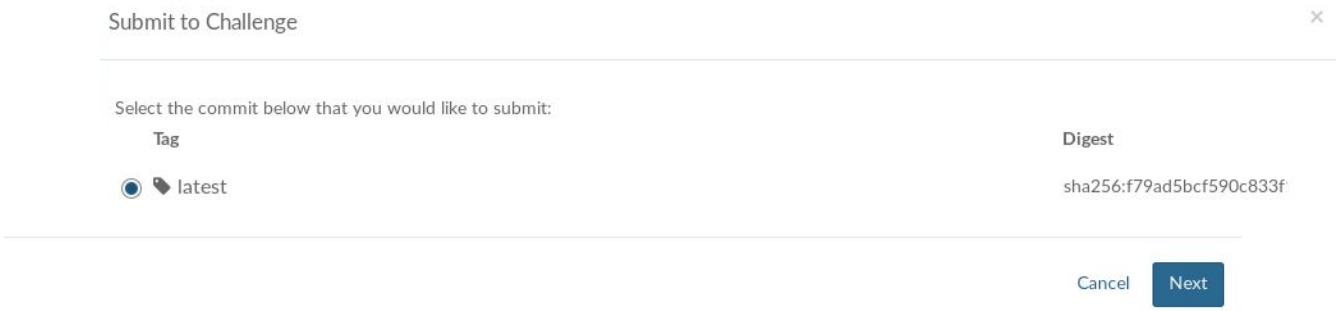


3 . Submit the docker image

1) Click on the docker image. In the next view, select “Docker Repository Tools → Submit Docker Repository to Challenge”



To submit the docker to the challenge, first click on the docker image you want to submit which brings you to the following screen:



For each of the three tasks there is an individual submission queue:

- MOOD 2020 - Sample-Level,
- MOOD 2020 - Pixel-Level,

Choose the task for which you want to submit and click “Next”.

If you plan to participate in **multiple tasks** of the challenge, you need to **submit your corresponding docker to each queue individually**.

Specify if you are entering alone or as a team. The submission pipeline will be checked in regular intervals. You will then be notified whether your submission is invalid (due to wrong format, etc…) or has been accepted.

The committee wishes you much success :-)