

HELMHOLTZ

MODEL ZOO

Hans Werners, Jennifer Ahrens,

Engin Eren, Philipp Heuser



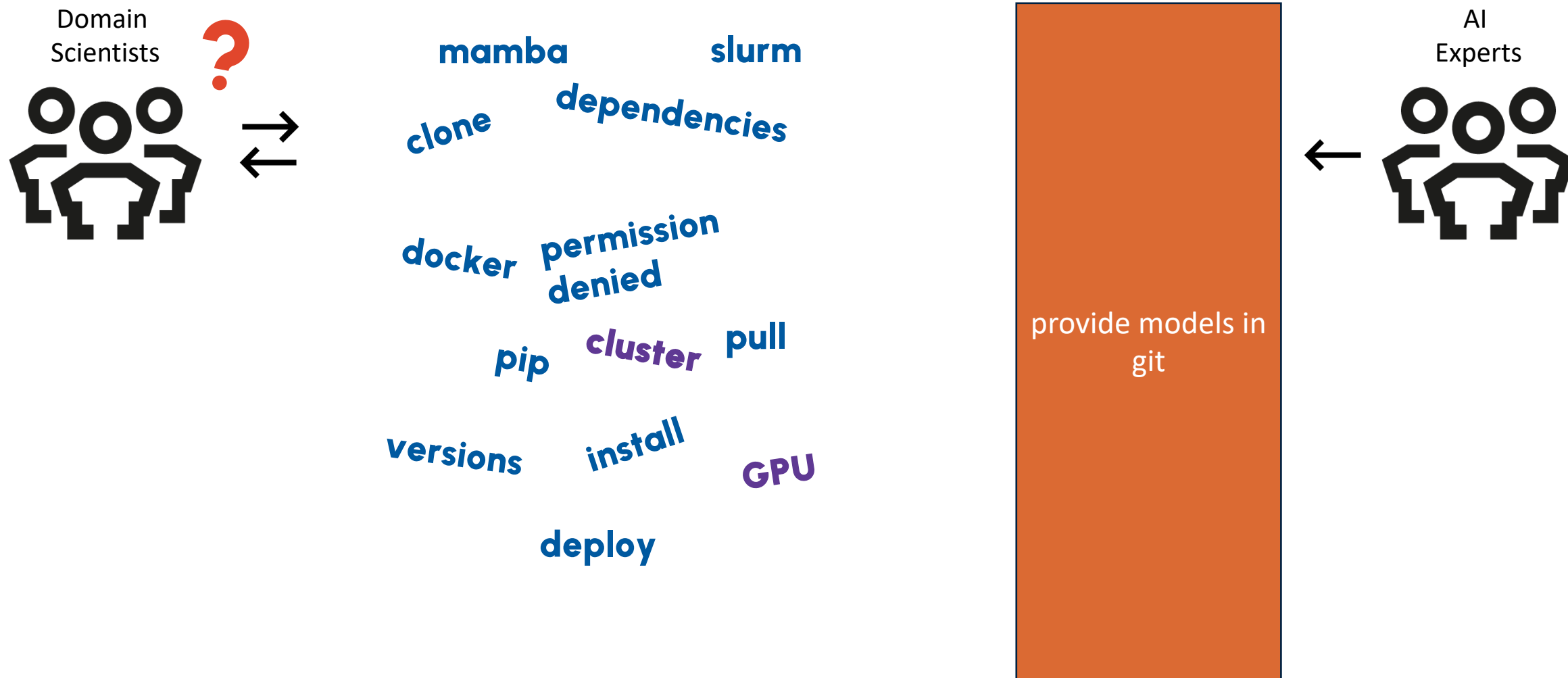
Helmholtz Model Zoo

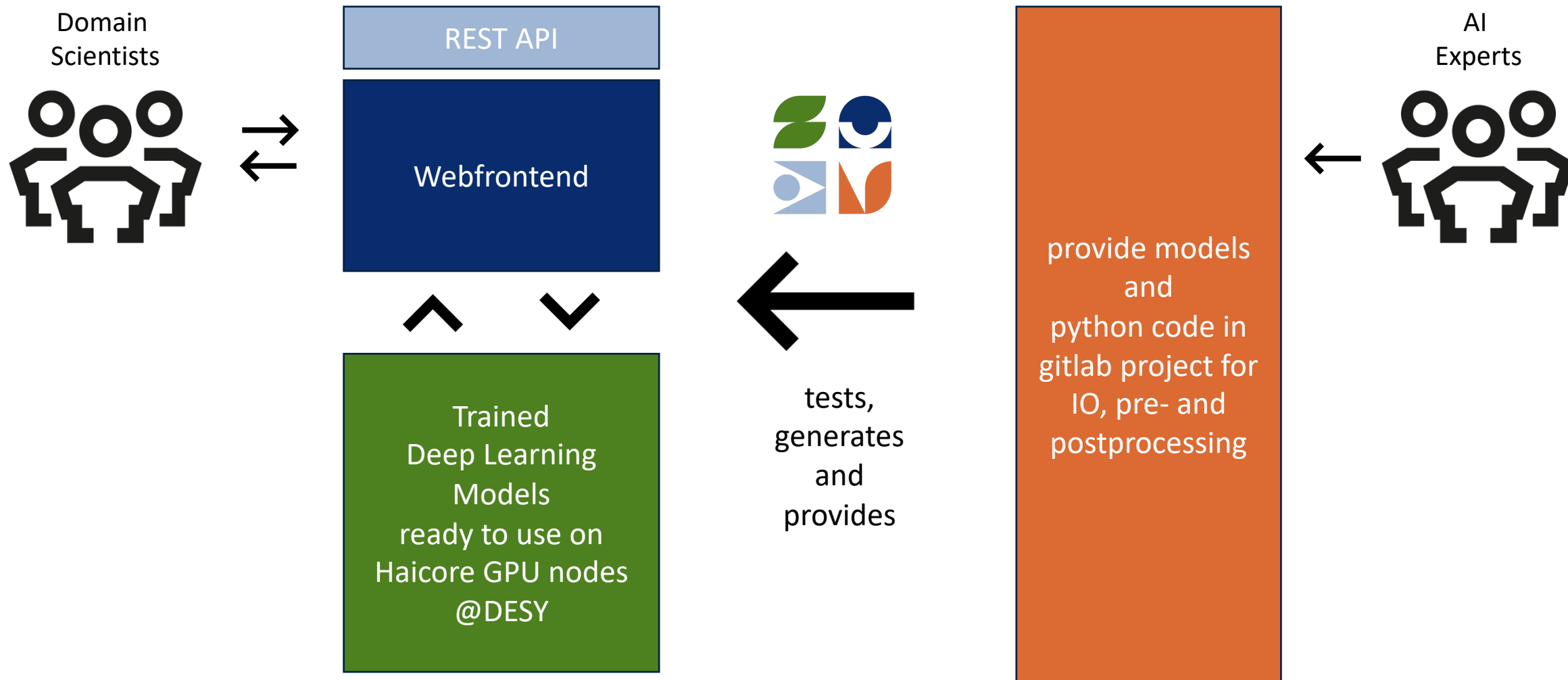
A Model Zoo is a collection of (pre)trained deep learning models, ready to use for inference.

These can include:

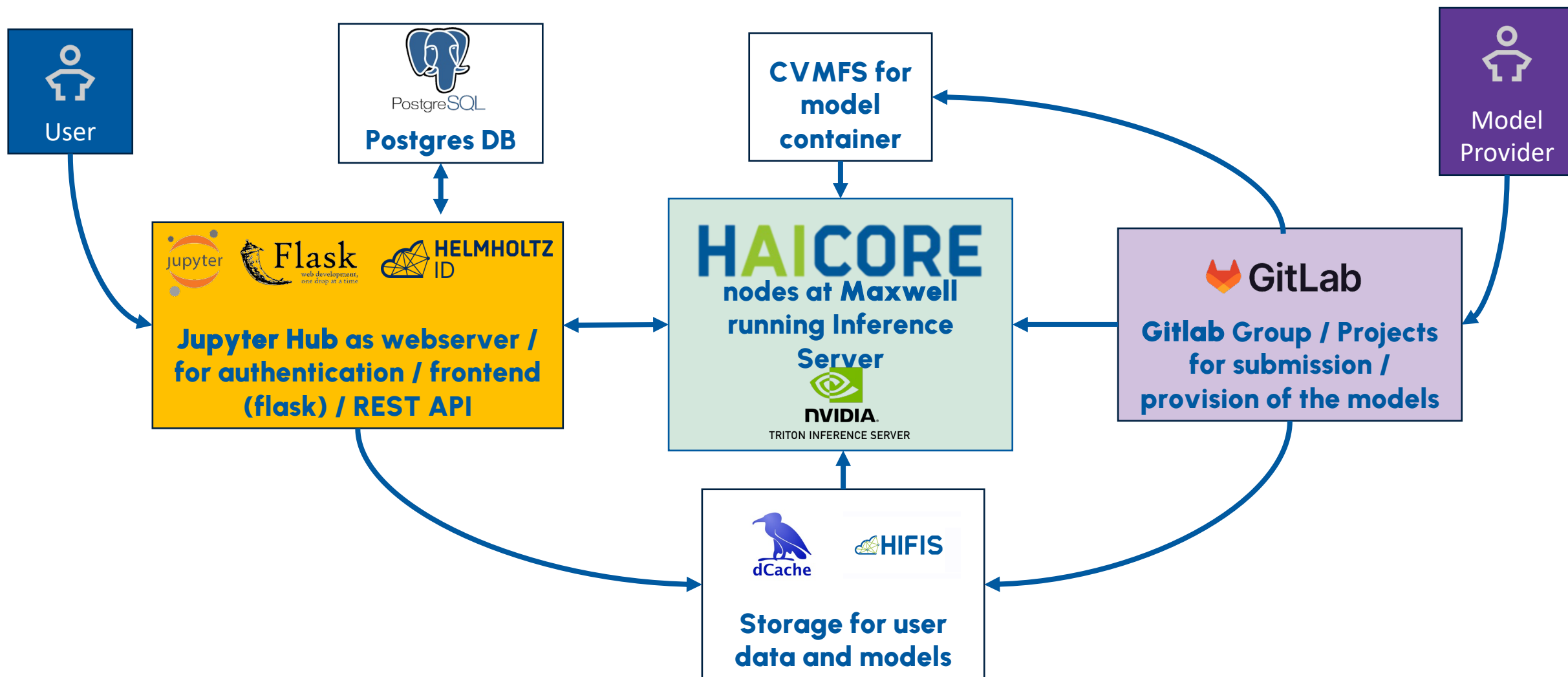
- specifically trained and fine-tuned models for defined tasks and data,
- domain and data type agnostic generalised models
- foundation models
- Your models!







Helmholtz Model Zoo



Why a “Helmholtz” Model Zoo?




Can't you just use xxx?

- **the data stays within Helmholtz**
- **flexible data-limit**
- **“private” models possible**
- **flexible usage (WebUI, REST-API, REST-API raw)**
- **free to use for Helmholtz**
- **support**
- **easy to use**
- **no installation hassle, no GPU hunting, no IT knowledge necessary**


Model Zoo for the User





HELMHOLTZ
MODEL ZOO

[Home](#)[Models](#)[My Jobs](#)[For Model Providers](#)[About](#)



HELMHOLTZ
MODEL ZOO

Pilot Phase (Beta Test)

The Helmholtz Model Zoo is currently only in the **Pilot Phase/Beta Test!**

What does this mean?

You are here as an early tester. The Model Zoo is generally speaking working. There are some features we are still working on. **The Model Zoo may undergo updates, changes and short interruptions at any time. (Sorry)**

Nevertheless, we welcome you here and ask for your input to make the Helmholtz Model Zoo a great tool for scientific endeavours. Please do not hesitate to provide your models and to test the website! We are looking forward to any bug report, comment or suggestion at hmz-admin@desy.de


Welcome to the Helmholtz Model Zoo. HMZ is offering scientific deep learning models for inference in the browser. Here you find the tools Helmholtz members trained to conduct the *research for grand challenges*.

The content of the Helmholtz Model Zoo is provided by experts from the association, and when you are affiliated to one of the Helmholtz Centers, you are invited to login with your institutional credentials and use them!

Start

For Model Providers

If you are a Machine Learning expert yourself, and are looking for a way to share your model in the Helmholtz Model Zoo, find a guide [here](#).



HELMHOLTZ
IMAGING

Capturing the
world of science.

HELMHOLTZ

[Imprint](#)[Privacy](#)[Declaration of Accessibility](#)

©2025 Deutsches Elektronen-Synchrotron DESY

Model overview

Selecting a model redirects you to the model-specific input-page.

Mnist Demo

by Helmholtz Model Zoo Team

This is a demo model for the model zoo. It is a simple MNIST model. Don't judge about its...

More →

inpainting-lama-onnx25

by Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silverstov, Naejin Kong, Harshith Goka, Kiwoong Park, Victor Lempitsky

Adaptation from HF --> <https://huggingface.co/Carve/LaMa-ONNX>

More →


Photon
Energy: 90 [GeV]
Event: 4
Time step: 0.97995 [ns]

CaloClouds

by Anatolii Korol

CaloClouds 3 simulates high-granularity photon showers using point cloud diffusion and...

More →



MODEL ZOO

Home

Models

My Jobs

For Model Providers

About

inpainting-lama-onnx25

by Roman Suvorov , Elizaveta Logacheva , Anton Mashikhin , Anastasia Remizova , Arsenii Ashukha , Aleksei Silverstrov , Naejin Kong , Harshith Goka , Kiwoang Park , Victor Lempitsky

Source repository of model

Publication:

Primary contact: hmz_admin@desy.de

License: Apache license 2.0

Tags: HMZ, HMZ Demo, inpainting

Model Description:

Adaptation from HF --> <https://huggingface.co/Carve/LaMa-ONNX>

This is a Fourier Convolutions, CNN etc.. model, trained on Images.

Data Input

This model expects as input:

Field Name	Help Text	Data Type expected
raw_image	Upload an image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'	image 2D (jpg, png)
mask	Upload the relative mask to the image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'	image 2D (jpg, png)


Result

As result you can expect:

Result Folder Name	Result Description	Result Filetype
output	This is inpainted image	image 2D (jpg , png)

When using this model, please cite:

Use Model →



HELMHOLTZ IMAGING

Capturing the world of science.

HELMHOLTZ

Imprint

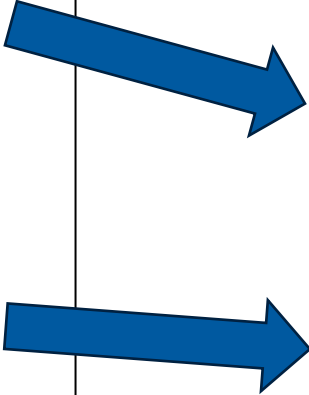
Privacy


Declaration of Accessibility

©2025 Deutsches Elektronen-Synchrotron DESY

How to use inpainting-lama-onnx25 with REST_API?
[See here.](#)

Model Zoo for the User



HELMHOLTZ
IMAGING

MODEL ZOO

[Home](#) [Models](#) [My Jobs](#) [For Model Providers](#) [About](#)

Input Form for inpainting- lama-onnx25


Description

Adaptation from HF --> <https://huggingface.co/Carve/LaMa-ONNX>

Upload Files

Select Files from Upload Directory

Upload an image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'




Drag and drop files here or click to select files

Accepted file types: jpg, png

Select Files

Upload the relative mask to the image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'




Drag and drop files here or click to select files

Accepted file types: jpg, png

Select Files

Submit

HELMHOLTZ
IMAGING

Capturing the
world of science.

HELMHOLTZ

[Imprint](#) [Privacy](#) [Declaration of Accessibility](#)

©2025 Deutsches Elektronen-Synchrotron DESY

Model Zoo for the User



HELMHOLTZ
IMAGING

MODEL ZOO

HomeModelsMy JobsFor Model ProvidersAbout

Job inpainting-lama-onnx25

Status
Model: inpainting-lama-onnx25
Input: /pnfs/desy.de/hifis-storage/HMZ/user/pheuser/upload/dataset_inpainting-lama-onnx25_2025-06-24_11-58-57

Status: COMPLETED
Start: 2025-06-24 11:58:58
End: 2025-06-24 11:59:04

Results / Download

Results
Download all result files as one zip:
output 1 files
inpainting-lama-onnx25_prediction_day1.jpg
ViewDownload

Logs

Here you can quickly scan some of the log files for information.

status_logslurm_outslurm_error

> This is the content of the logfile:
> /pnfs/desy.de/hifis-storage/HMZ/user/pheuser/result/result_inpainting-lama-onnx25_70790f15/logs/status_log.log

Done

Please cite:

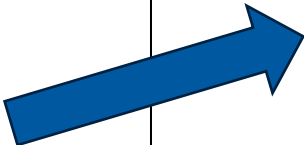
HELMHOLTZ
IMAGING

Capturing the
world of science.

HELMHOLTZ


ImprintPrivacyDeclaration of Accessibility

©2025 Deutsches Elektronen-Synchrotron DESY



Model Zoo for the User



 **MODEL ZOO**

HomeModelsMy JobsFor Model ProvidersAbout

Input Form for CaloClouds


Description

CaloClouds 3 simulates high-granularity photon showers using point cloud diffusion and normalizing flows, matching Geant4 accuracy with faster performance. Optimized for full detector use and reliable physics reproduction.

Upload Files

Select Files from Upload Directory

Provide txt config file.




Drag and drop files here or click to select files

Accepted file types: txt

Select Files

Submit

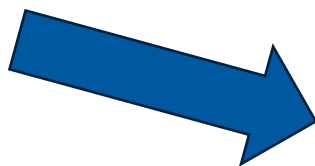
 **HELMHOLTZ IMAGING**

Capturing the world of science.

HELMHOLTZ

ImprintPrivacyDeclaration of Accessibility

```
input.txt
Energy=50
eta=60
phi=0
N=1000
```



Model Zoo for the User



Job CaloClouds

Status

Model: caloclouds

Input: /pnfs/desy.de/hifis-storage/HMZ/user/eren/upload/
dataset_caloclouds_2025-07-03_09-46-11

Status: COMPLETED ✓

Start: 2025-07-03 09:46:13
End: 2025-07-03 09:49:08

Results / Download

Results

Download all result files as one zip: [\[1\]](#)



Logs

Here you can quickly scan some of the log files for information.



```
> This is the content of the logfile:
> /pnfs/desy.de/hifis-storage/HMZ/user/eren/result/result_caloclouds_c293fd2a/logs/status_log.log
-----
Done
```

Please cite:
<https://arxiv.org/pdf/2309.05704>

```
[1]: import uproot

file = uproot.open("caloclouds_prediction_input.root")

[2]: print(file.keys())

['Events;1']
```

```
[13]: data[2:10]

[13]: [{noutput: 8330, output: [6, 13, ..., 9, 0.287]},
      {noutput: 8882, output: [9, 19, ..., 3, 0.166]},
      {noutput: 8426, output: [24, 48, ..., 11, 0.382]},
      {noutput: 8990, output: [8, 33, ..., 6, 0.253]},
      {noutput: 8394, output: [10, 12, ..., 7, 0.579]},
      {noutput: 8722, output: [11, 12, ..., 10, 0.343]},
      {noutput: 8470, output: [15, 31, ..., 9, 0.377]},
      {noutput: 8614, output: [7, 23, ..., 9, 0.0446]}]
```

```
-----
backend: cpu
nbytes: 1.0 MB
type: 8 * {
  noutput: int32,
  output: var * float64
}
```

Model Zoo for the User



Job CaloClouds

Status

Model: caloclouds

Input: /pnfs/desy.de/hifis-storage/HMZ/user/eren/upload/
dataset_caloclouds_2025-07-03_09-46-11

Status: COMPLETED ✓

Start: 2025-07-03 09:46:13
End: 2025-07-03 09:49:08

Results / Download

Results

Download all result files as one zip: [\[1\]](#)



Logs

Here you can quickly scan some of the log files for information.



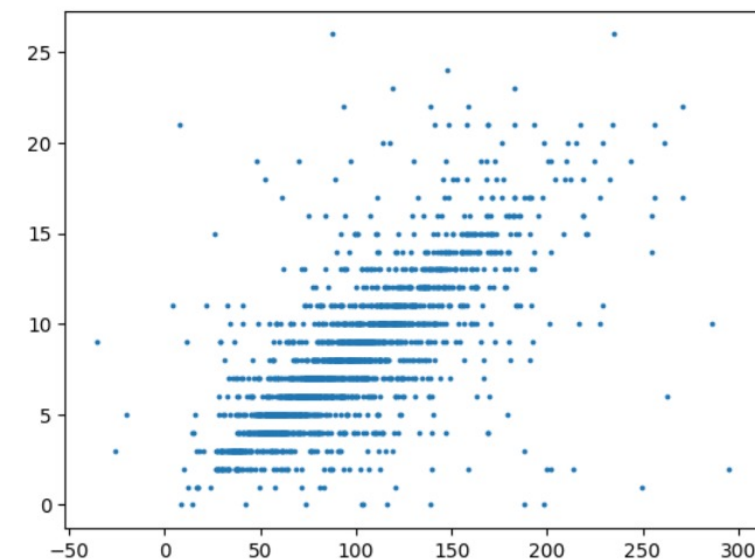
```
> This is the content of the logfile:  
> /pnfs/desy.de/hifis-storage/HMZ/user/eren/result/result_caloclouds_c293fd2a/logs/status_log.log  
-----  
Done
```

Please cite:
<https://arxiv.org/pdf/2309.05704>

```
[1]: import uproot  
  
file = uproot.open("caloclouds_prediction_input.root")  
  
[2]: print(file.keys())  
  
['Events;1']
```

```
[22]: import matplotlib.pyplot as plt  
plt.scatter(x,z,s=3)
```

```
[22]: <matplotlib.collections.PathCollection at 0x148de86817d0>
```



Who can use the models?

Only colleagues from Helmholtz can use it (and members of a Helmholtz ID VO)

How can I run an inference?

- Use the webinterface.**
- Use a REST API**

What is about my data (user)?

- You upload the data to a folder owned by you in dcache via the webinterface.**
- You own your data, and it does not leave Helmholtz.**
- You can also use the documented ways to upload data to dcache via rclone, webdav, etc.
(recommended for large datasets)**
- You can upload a reasonable amount of data (probably not PB, but a few TB could be fine).**
- We will delete all your data after 3 month, or so? (Not entirely defined yet)**

For model providers



Who can provide models?

All scientists in Helmholtz can login via the Helmholtz ID and provide their models.

What kind of models?



ONNX



PyTorch



TensorFlow

OpenVINO™



Which hardware?

Our nodes are provided by HAICORE, run at DESY, each equipped with 4 Nvidia L40S.



Can I make my model “private”?

Yes, you will be able to give the usernames of people who are entitled to use your model? (Later via a Helmholtz ID VO)

What do I have to do?

- We provide you with a template in the (DESY)-Gitlab.
- We try to get as much as possible of your shoulders, keeping a maximum of flexibility
- You have to add your model and implement the necessary for pre- and postprocessing
- You have to fill in a form for the necessary meta data (some technical details, description, publication)

Models / config.pbtxt



Before a model can be served by Triton Inference Server, it must be converted into a backend-specific format that Triton can load and run. This step is called exporting the model.

```
import torch
import torch.nn as nn

class MyModel(nn.Module):
    def __init__(self):
        super().__init__()
        self.fc = nn.Linear(10, 5)

    def forward(self, x):
        return self.fc(x)

model = MyModel()
example_input = torch.randn(1, 10)

# Scripted (preferred for dynamic models)
scripted_model = torch.jit.script(model)
scripted_model.save("model/1/your_model_name_model.pt")
```

Models / config.pbtxt



```
name: "Hans_mnist"
platform: "pytorch_libtorch"
max_batch_size : 0
input [
  {
    name: "input__0"
    data_type: TYPE_FP32
    dims: [ 1, 28, 28 ]
    reshape { shape: [ 1, 1, 28, 28 ] }
  }
]
output [
  {
    name: "output__0"
    data_type: TYPE_FP32
    dims: [ 1, 10]
    reshape { shape: [ 1, 10 ] }
  }
]
```

*** a draft for the
config.pbtxt**

Model Zoo for the Model Provider



HELMHOLTZ
IMAGING
MODEL ZOO

HomeModelsMy JobsFor Model ProvidersAbout

Add a new model to Helmholtz Model Zoo

Disclaimer

Before you start creating a new model, please read! By creating a new model repository here, you agree to the following:

I agree that Helmholtz Imaging offers this model and the provided routines for pre- and postprocessing as a service for all colleagues of any of the Helmholtz Centers, independent of the actual license chosen for this model. For the usage of the Helmholtz Model Zoo the rules as listed in the [Terms of Use](#) for Helmholtz cloud services apply.

Helmholtz Imaging will operate the Helmholtz Model Zoo with best effort, but without any guarantee. Operation of the Helmholtz Model Zoo is without warranty of any kind, express or implied, including but not limited to the warrants of merchantability, fitness for particular purpose and noninfringement. In no event shall Helmholtz Imaging be liable for any claim, damages or other liability, whether in action of contract, tort or otherwise, arising from, out or in connection with the model and software, or the use or other dealings in the service.

I confirm that I hold the right to provide this model and it's accompanying files (images, templates etc.) to the colleagues of any of the Helmholtz Centers, and that any linked or imported software and libraries have a sufficiently permissive license.

I confirm that the provided model has only been trained on data, which allows for the distribution of this model.

Helmholtz Imaging will show/link the licenses you select for your contribution on the Helmholtz Model Zoo web interface. Helmholtz Imaging will not enforce your license.

You can restrict the access to the model to individual colleagues by providing their usernames. This restricts only the access to the model itself, it is the file with the trained weights. The functions you provide for pre- or postprocessing are still visible for anyone! Therefore you can use a more restrictive license for the model, but not for any functions you need around it!

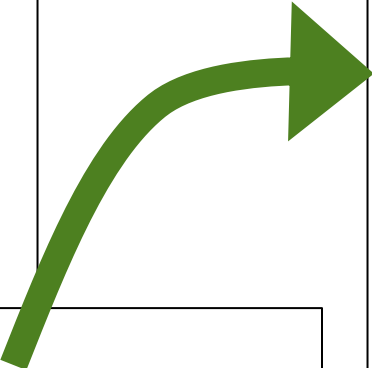
For handling the containers with the pre- and postprocessing functions we are using [CernVM-FS](#). This implied that anyone who can mount the CernVM-FS, can get access to the container images containing your pre- and postprocessing functions, but not the model files.

How do I add a model?

Contributed models must live in the DESY gitlab and follow a very specific structure to work within the modelzoo. To make things easier for you, we provide you with a model template, you only need to fill out. We also provide tests, which must succeed, in order for your model to be deployed.

To get started, follow these steps:

- 1) [Find out your DESY gitlab user ID.](#)
- 2) Fill out [this form](#).
- 3) You get an email with the link to the new project repository, where everything is already set up for you.
- 4) Generate the meta-data json file for your project. Most of the information we need to compile the website and to get as much coding as possible of your



Create a repository for you model

To create a new git project for HMZ, you must be known to the [DESY gitlab](#). Therefore, please log in at least once with your Helmholtz ID to the DESY gitlab. Once logged in, go to [user profile](#), under 'Main Settings' you will find your gitlab User ID, which you need here.

model name

my_new_model

Name of new HMZ Model Project

gitlab user id

xxx

Your DESY gitlab user ID

Submit

M

my_new_model

main

my_new_model

+

Find file

Edit

Code

Merge branch 'philipp.heuser-main-patch-99074' into 'main'

...

d2648ff9

History

Name	Last commit	Last update
.gitlab/ci	updating CI Pipeline to be build on main...	2 days ago
example_data	Documentation	5 months ago
images	copy meta json in the working dir of the...	3 days ago
model	postprocessor template	2 weeks ago
server_files	change to not count the mask and mak...	2 days ago
tests	Created template structure	8 months ago
.gitignore	adding pre-commit file	4 months ago
.gitlab-ci.yml	copy meta json in the working dir of the...	3 days ago
.pre-commit-config.yaml	cleaned_inference.py	4 months ago
Dockerfile	copy meta json in the working dir of the...	3 days ago
README.md	adding pre-commit file	4 months ago
new_model.md	Edit new_model.md	10 hours ago

Next time...
More to show!!!



my_new_model

main

my_new_model

Merge branch 'Philipp.Heuser-main-patch-99074' into 'main'

Philipp Heuser authored 10 hours ago

Name	Last commit
..	
__init__.py	general architecture idea of the temp...
howto_inference.md	changes for links and requirements ;
howto_postprocess.md	changes for links and requirements ;
howto_preprocess.md	updated the self.valid_extensions_di
inference.py	changes for links and requirements ;
postprocessor.py	changes for links and requirements ;
preprocessor.py	change to not count the mask and m
requirements_model.txt	first version of Pipelein to build Mod
server_files	change to not count the mask and mak... 2 days ago
tests	Created template structure 8 months ago
.gitignore	adding pre-commit file 4 months ago
.gitlab-ci.yml	copy meta json in the working dir of the 3 days ago
.pre-commit-config.yaml	
Dockerfile	
README.md	
new_model.md	

Name	Last commit
..	
__init__.py	general architecture idea of the temp...
howto_inference.md	changes for links and requirements ;
howto_postprocess.md	changes for links and requirements ;
howto_preprocess.md	updated the self.valid_extensions_di
inference.py	changes for links and requirements ;
postprocessor.py	changes for links and requirements ;
preprocessor.py	change to not count the mask and m
requirements_model.txt	first version of Pipelein to build Mod

HowTo preprocessor.py

HowTo inference.py

HowTo postprocess.py

	Last update
	7 months ago
	1 week ago
	1 week ago
	2 days ago
	1 week ago
	1 week ago
	2 days ago
	5 months ago

preprocessor.py

- read file from disk
- normalise, reshape etc.

inference.py

- send numpy arrays to triton
- receive results from triton

postprocess.py

- transform model output to human readable
- save files to disk.

Pilot Phase (Beta Test)

The Helmholtz Model Zoo is currently only in the **Pilot Phase/Beta Test!**

What does this mean?

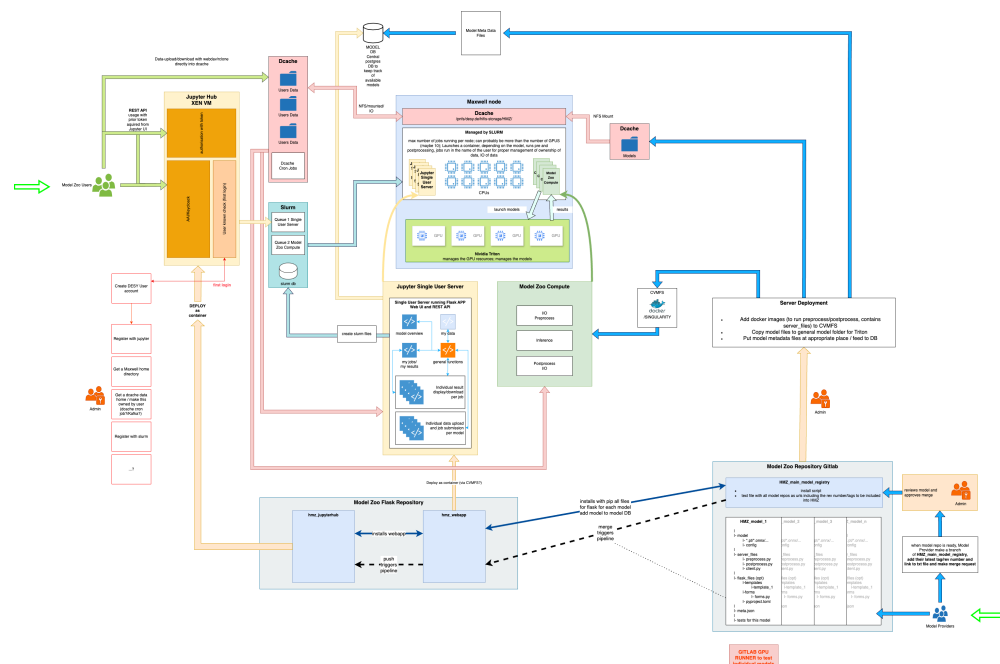
You are here as an early tester. The Model Zoo is generally speaking working. There are some features we are still working on. **The Model Zoo may undergo updates, changes and short interruptions at any time. (Sorry)**

Nevertheless, we welcome you here and ask for your input to make the Helmholtz Model Zoo a great tool for scientific endeavours. Please do not hesitate to provide your models and to test the website! We are looking forward to any bug report, comment or suggestion at hmz-admin@desy.de

Preliminary URL:
<https://hmz-hub.desy.de>

Get in touch
hmz-admin@desy.de
& talk to us here

Helmholtz Model Zoo



HAICORE



HELMHOLTZ
IMAGING

Thanks to:

Christian Voss
Elena Gapon
Frank Schlünzen
Johannes Reppin
Kars Ohrenberg
Lusine Yakovleva
Moritz Hamming
Patrick Fuhrmann
Sven Sternberger
Thomas Hartmann
Tim Wetzel
Uwe Jandt
Yves Kemp
Alexander Trautsch
Stefan Bujack

Talk to us!



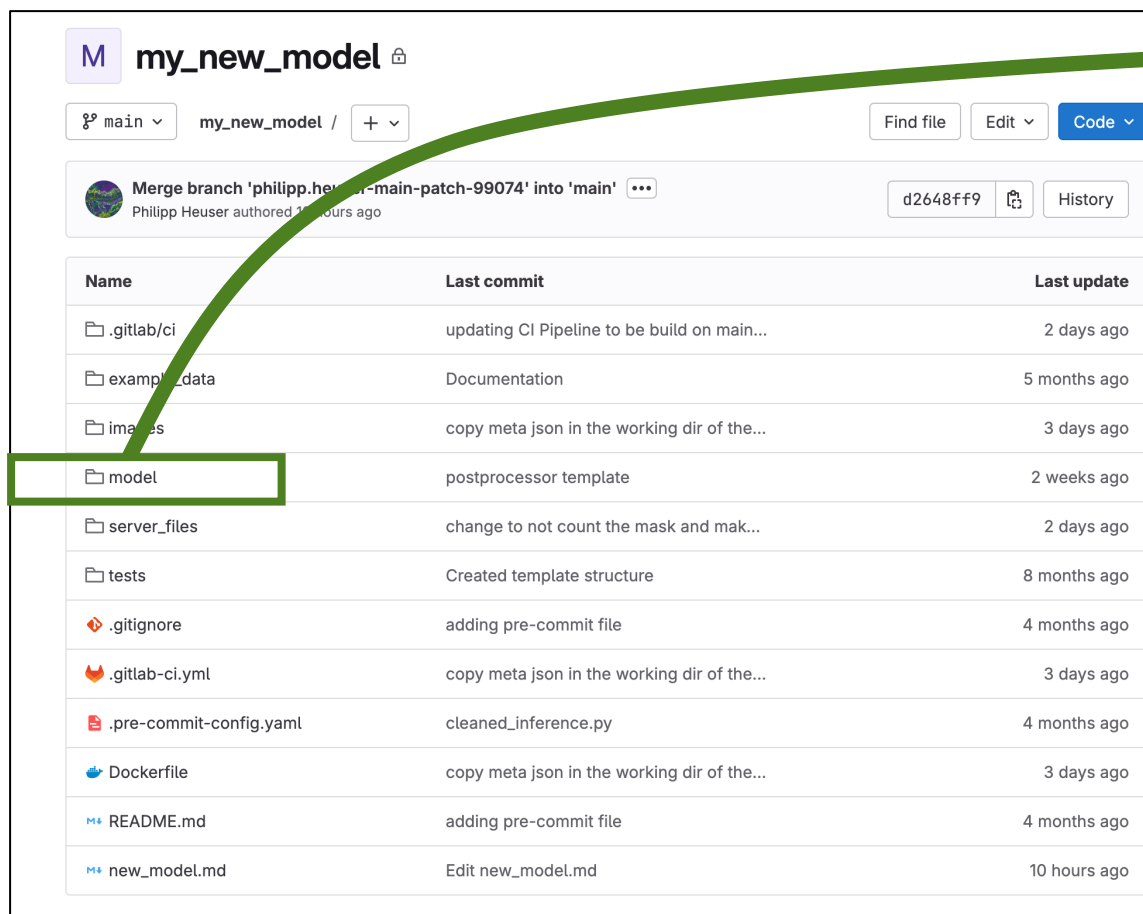
HELMHOLTZ
MODEL ZOO



support@helmholtz-imaging.de

Model Zoo for the Model Provider

model folder



my_new_model

main my_new_model / +

Find file Edit Code

Merge branch 'philipp.heuser/main-patch-99074' into 'main' d2648ff9 History

Name	Last commit	Last update
.gitlab/ci	updating CI Pipeline to be build on main...	2 days ago
example_data	Documentation	5 months ago
images	copy meta json in the working dir of the...	3 days ago
model	postprocessor template	2 weeks ago
server_files	change to not count the mask and mak...	2 days ago
tests	Created template structure	8 months ago
.gitignore	adding pre-commit file	4 months ago
.gitlab-ci.yml	copy meta json in the working dir of the...	3 days ago
.pre-commit-config.yaml	cleaned_inference.py	4 months ago
Dockerfile	copy meta json in the working dir of the...	3 days ago
README.md	adding pre-commit file	4 months ago
new_model.md	Edit new_model.md	10 hours ago

1

model.pt/onnx/...

config.pbtxt

```
name: "Hans_mnist"
platform: "pytorch_libtorch"
max_batch_size : 0
input [
  {
    name: "input__0"
    data_type: TYPE_FP32
    dims: [ 1, 28, 28 ]
    reshape { shape: [ 1, 1, 28, 28 ] }
  }
]
output [
  {
    name: "output__0"
    data_type: TYPE_FP32
    dims: [ 1, 10]
    reshape { shape: [ 1, 10 ] }
  }
]
```


Model Zoo for the Model Provider

server_files folder



M

my_new_model

main

my_new_model

Merge branch 'philipp.heuser-main-patch-99074' into 'main'

Philipp Heuser authored 10 hours ago

Name	Last commit
.gitlab/...	updating CI Pipeline to be build on ma...
examples_data	Documentation
images	copy meta json in the working dir of th...
model	postprocessor template
server_files	change to not count the mask and mak...
tests	Created template structure
.gitignore	adding pre-commit file
.gitlab-ci.yml	copy meta json in the working dir of the...
.pre-commit-config.yaml	cleaned_inference.py
Dockerfile	copy meta json in the working dir of the...
README.md	adding pre-commit file
new_model.md	Edit new_model.md

Name	Last commit	Last update
..		
__init__.py	general architecture idea of the template	7 months ago
howto_inference.md	changes for links and requirements and dcache note	1 week ago
howto_postprocess.md	changes for links and requirements and dcache note	1 week ago
howto_preprocess.md	updated the self.valid_extensions_dict to handle multiple inputs	2 days ago
inference.py	changes for links and requirements and dcache note	1 week ago
postprocessor.py	changes for links and requirements and dcache note	1 week ago
preprocessor.py	change to not count the mask and make the difference clearer	2 days ago
requirements_model.txt	first version of Pipelein to build Model_Images	5 months ago

Model Zoo for the Model Provider



server_files folder

my_new_model

main my_new_model

Merge branch 'Philipp.heuser-main-patch-99074' into 'main'

Philipp Heuser authored 10 hours ago

Name	Last commit	Last update
..		
__init__.py	general architecture idea of the template	7 months ago
howto_inference.md	changes for links and requirements and dcache note	1 week ago
howto_postprocess.md	changes for links and requirements and dcache note	1 week ago
howto_preprocess.md	updated the self.valid_extensions_dict to handle multiple inputs	2 days ago
inference.py	changes for links and requirements and dcache note	1 week ago
postprocessor.py	changes for links and requirements and dcache note	1 week ago
preprocessor.py	change to not count the mask and make the difference clearer	2 days ago
requirements_model.txt	first version of Pipeline to build ModelImages	5 months ago

server_files

Name	Last commit	Last update
.gitlab/	updating CI Pipeline to be build on ma	
examples_data	Documentation	
images	copy meta json in the working dir of th	
model	postprocessor template	
server_files	change to not count the mask and mak...	2 days ago
tests	Created template structure	8 months ago
.gitignore	adding pre-commit file	4 months ago
.gitlab-ci.yml	copy meta json in the working dir of the	3 days ago
.pre-commit-config.yaml		
Dockerfile		
README.md		
new_model.md		

preprocessor.py

- read file from disk
- normalise, reshape etc.

inference.py

- send numpy arrays to triton
- receive results from triton

postprocess.py

- transform model output to human readable
- save files to disk.

Model Zoo for the Model Provider

server_files folder



my_new_model

main my_new_model

Merge branch 'Philipp.heuser-main-patch-99074' into 'main'

Name	Last commit
.gitlab/	updating CI Pipeline to be build on ma
examples_data	Documentation
images	copy meta json in the working dir of th
model	postprocessor template
server_files	change to not count the mask and mak...
tests	Created template structure
.gitignore	adding pre-commit file
.gitlab-ci.yml	copy meta json in the working dir of the
.pre-commit-config.yaml	
Dockerfile	
README.md	
new_model.md	

Name	Last commit
..	
__init__.py	general architecture idea of the tem
howto_inference.md	changes for links and requirements
howto_postprocess.md	changes for links and requirements
howto_preprocess.md	updated the self.valid_extensions_di
inference.py	changes for links and requirements
postprocessor.py	changes for links and requirements
preprocessor.py	change to not count the mask and m
requirements_model.txt	first version of Pipeline to build Mod

preprocessor.py

- read file from disk
- normalise, reshape etc.

inference.py

- send numpy arrays to triton
- receive results from triton

HowTo preprocessor.py

py

model output
adable
disk.

2025 | HELMHOLTZ IMAGING |

Model Zoo for the Model Provider



M my_new_model


main

my_new_model / +

Find file


Edit

Code













 Merge branch 'philipp.heuser-main-patch-99074' into 'main'

Philipp Heuser authored 10 hours ago

d2648ff9



History

Name	Last commit	Last update
 .gitlab/ci	updating CI Pipeline to be build on main...	2 days ago
 example_data	Documentation	5 months ago
 images	copy meta json in the working dir of the...	3 days ago
 model	postprocessor template	2 weeks ago
 server_files	change to not count the mask and mak...	2 days ago
 tests	Created template structure	8 months ago
 .gitignore	adding pre-commit file	4 months ago
 .gitlab-ci.yml	copy meta json in the working dir of the...	3 days ago
 .pre-commit-config.yaml	cleaned_inference.py	4 months ago
 Dockerfile	copy meta json in the working dir of the...	3 days ago
 README.md	adding pre-commit file	4 months ago
 new_model.md	Edit new_model.md	10 hours ago

example data:
data to test the model

Model Zoo for the Model Provider

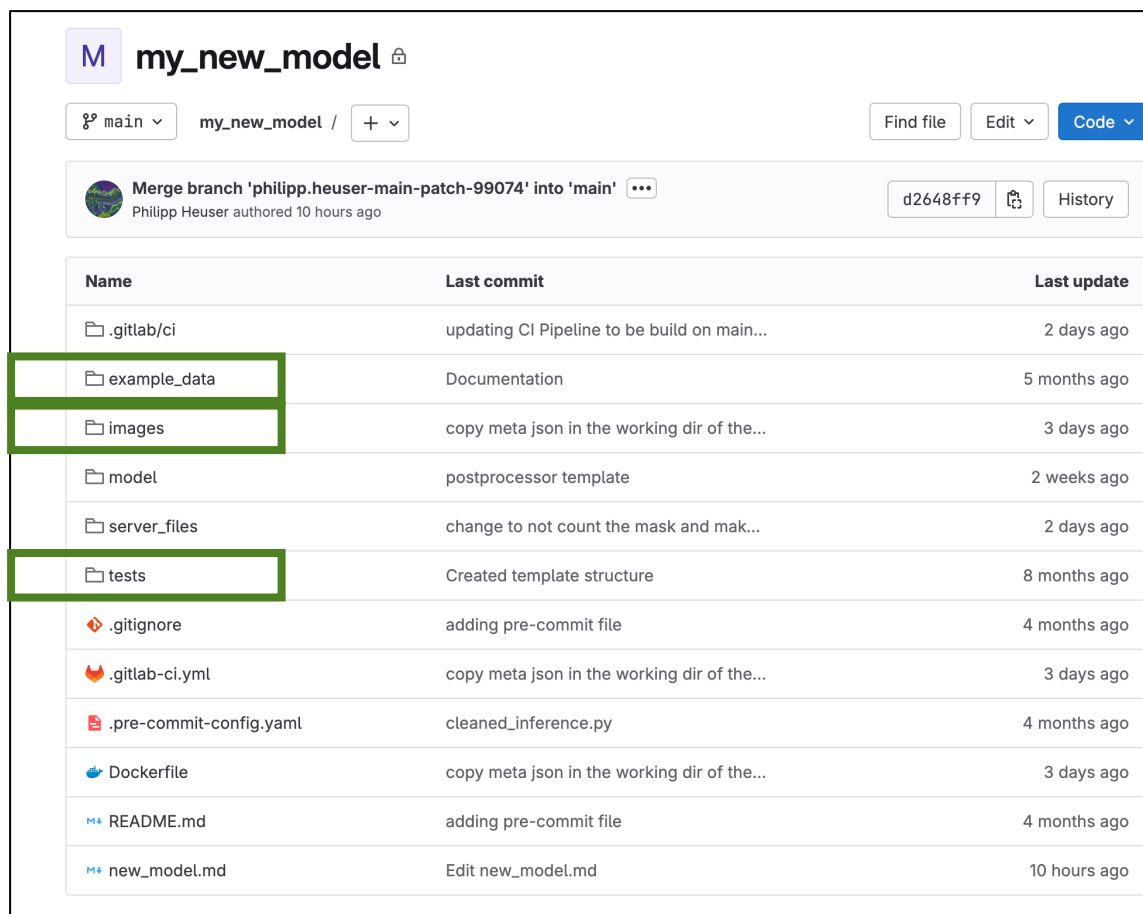














Name	Last commit	Last update
.gitlab/ci	updating CI Pipeline to be build on main...	2 days ago
example_data	Documentation	5 months ago
images	copy meta json in the working dir of the...	3 days ago
model	postprocessor template	2 weeks ago
server_files	change to not count the mask and mak...	2 days ago
tests	Created template structure	8 months ago
.gitignore	adding pre-commit file	4 months ago
.gitlab-ci.yml	copy meta json in the working dir of the...	3 days ago
.pre-commit-config.yaml	cleaned_inference.py	4 months ago
Dockerfile	copy meta json in the working dir of the...	3 days ago
README.md	adding pre-commit file	4 months ago
new_model.md	Edit new_model.md	10 hours ago

example data:
data to test the model

images:
image(s) for illustration of website

Model Zoo for the Model Provider



Name	Last commit	Last update
 .gitlab/ci	updating CI Pipeline to be build on main...	2 days ago
 example_data	Documentation	5 months ago
 images	copy meta json in the working dir of the...	3 days ago
 model	postprocessor template	2 weeks ago
 server_files	change to not count the mask and mak...	2 days ago
 tests	Created template structure	8 months ago
 .gitignore	adding pre-commit file	4 months ago
 .gitlab-ci.yml	copy meta json in the working dir of the...	3 days ago
 .pre-commit-config.yaml	cleaned_inference.py	4 months ago
 Dockerfile	copy meta json in the working dir of the...	3 days ago
 README.md	adding pre-commit file	4 months ago
 new_model.md	Edit new_model.md	10 hours ago

example data:
data to test the model

images:
image(s) for illustration of website

tests:
test files to check your code.

Model Zoo for the Model Provider

meta.json



Generate your meta data json

Based on the information here, the WebUI/APIs are generated, and various git pipelines controlled. Please provide the 'ground truth' of your information here. You can always edit the respective files locally. Please make sure that the provided info here are consistent with whatever you do in the pre-/postprocessing and with your model.

Here we do not store the provided information, it is just a helper for you to create the required files. Please download and place the generated files at the respective path in the git project for your model. You can always upload a previously generated file, and edit it again.

Start changing an existing meta data file, by uploading it.

Upload your own data

Browse...

No file selected.

Enter meta data for your model.

MODEL_BASE_INFO

MODEL_CONFIG

DATA_IN

DATA_OUT

EXTRA_INPUT

ADMIN_INFO

CONNECT

Basic Information about the model:

Name

Name of the model identical to the name of the git_repository.

Display_Name

Name of the model as e.g. displayed in the webfrontend

Description

Important! Free text description of the model e.g. displayed in the webfrontend. This text makes the user want to use your model!

Python_Version

3:12

The python version with which your model needs to run

Model_Version

The version of your model

Model_Type

Which AI-model was trained? Here you can indicate the type/name/architecture of the network, if any. e.g. U-Net, Resnet-50 ...

Trained_On

Data used for training the model. Here you can indicate the name or a link to the dataset used for training.

License

License of the Model

git_Repo

Link to the original git repo of the network, as used for training (strongly recommended, but optional)

Authors

People involved and to be referenced for this model (optional) (Comma separated string)

Primary_Contact

Email address for primary contact / corresponding author

Publication

DOI for a publication to be referenced by the users. (optional)



- The meta.json is the heart of the HMZ,
- here we collect the information to render the website,
 - to define the model submission,
 - to create folders for IO,
 - etc.

Please use the form on the model zoo website to create the *meta.json* and place it on the toplevel of your model repo.

Helmholtz_Centers

Helmholtz Center(s) Involved

+ Add

Tags

How to input web-interface



HELMHOLTZ IMAGING

MODEL ZOO

Home

Generate your meta data js

Based on the information here, the WebUI/APIs are generated, and various git pipelines controlled. You can always edit the respective files locally. Please make sure that the provided info here are correct and with your model.

Here we do not store the provided information, it is just a helper for you to create the required files respective path in the git project for your model. You can always upload a previously generated file. Start changing an existing meta data file, by uploading it.

Upload your own data

Browse...

No file selected.

Enter meta data for your model.

MODEL_BASE_INFO

MODEL_CONFIG

DATA_IN

DATA_OUT

ADMIN_INFO

CONNECTIONS

Input Field 0

Input_Field_Name

Name for the field/parameter (only a-z, 0-9, _ and - allowed).

Input_Field_Help_Text

Please provide a help text for the users describing the parameter, and its recommended default value (limits/options and maybe example).

Input_Type

number

text

check_box_list

radio_button_list

file_upload

Default

For number, text, check_box_list or radio_button_list: Please provide the default value for your input field.

IO_Data_Type

text

For file upload: Please indicate the datatype to be provided/expected

File Extension

For file upload: Please provide the valid file extensions (e.g. .jpg, .tiff, .txt). If more than one is allowed, click ADD.

+ Add

Options

For radio or checkbox select lists: In case of radio or checkbox select lists, please provide a comma separated list of options.

+ Add

Download JSON

Download config .txt

Name for the field/parameter (only a-z, 0-9, _ and - allowed).

Input_Field_Help_Text

Please provide a help text for the users describing the parameter, and its recommended default value (limits/options and maybe example).

Input_Type

number

text

check_box_list

radio_button_list

file_upload

For number, text, check_box_list or radio_button_list: Please provide the default value for your input field.

IO_Data_Type

text

For file upload: Please indicate the datatype to be provided/expected

File Extension

HELMHOLTZ IMAGING

Capturing the world of science.

HELMHOLTZ

Imprint

Privacy

Declaration of Accessibility

© 2018 Deutsche Elektronen-Synchrotron DESY

Model Zoo for the User



Input Form for Mnist Demo


Description

This is a demo model for the model zoo. It is a simple MNIST model. Don't judge about its performance, its only purpose is to show the functionality of the Helmholtz Model Zoo.

Upload Files

Select Files from Upload Directory

images of handwritten numbers



Drag and drop files here or click to select files

Accepted file types: jpg, png

Select Files

Submit

Input Form for inpainting-lama-onnx25


Description

Adaptation from HF --> <https://huggingface.co/Carve/LaMa-ONNX>

Upload Files

Select Files from Upload Directory

please put 2D raw image here




Drag and drop files here or click to select files

Accepted file types: jpg, png

Select Files

please put 2D mask here



Drag and drop files here or click to select files

Accepted file types: jpg, png

Select Files

Submit

Models / config.pbtxt



Before a model can be served by Triton Inference Server, it must be converted into a backend-specific format that Triton can load and run. This step is called exporting the model.

```
import torch
import torch.nn as nn

class MyModel(nn.Module):
    def __init__(self):
        super().__init__()
        self.fc = nn.Linear(10, 5)

    def forward(self, x):
        return self.fc(x)

model = MyModel()
example_input = torch.randn(1, 10)

# Scripted (preferred for dynamic models)
scripted_model = torch.jit.script(model)
scripted_model.save("model/1/your_model_name_model.pt")
```

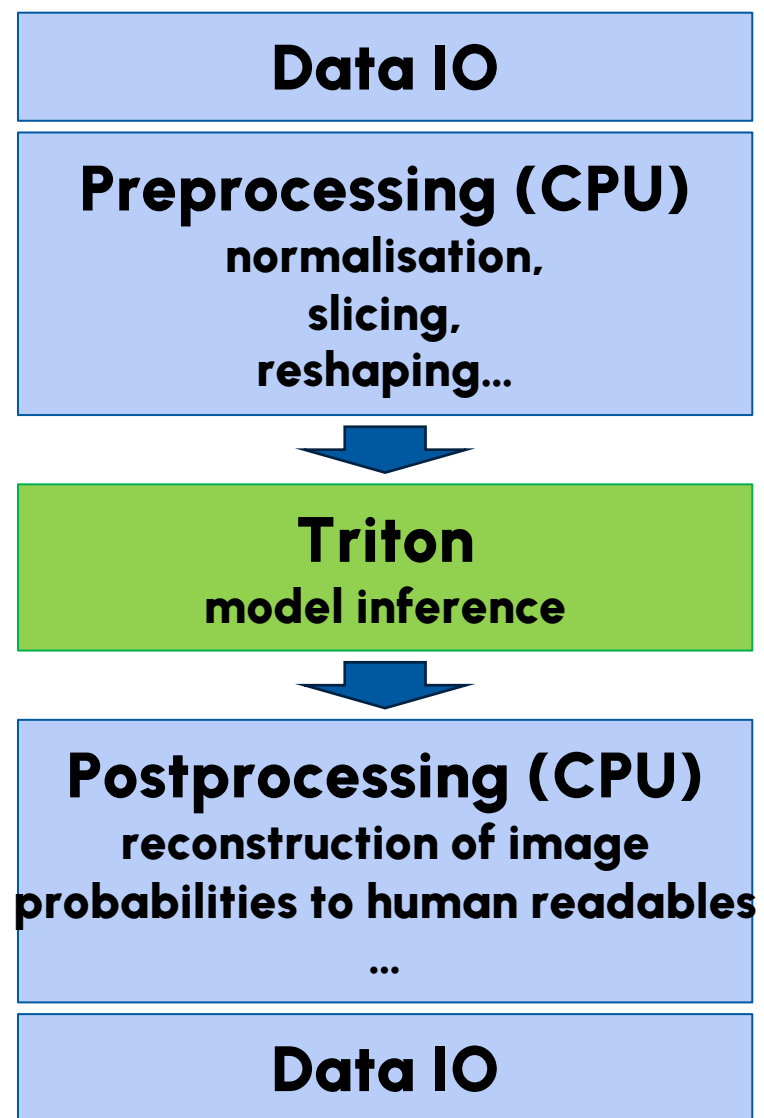
Models / config.pbtxt



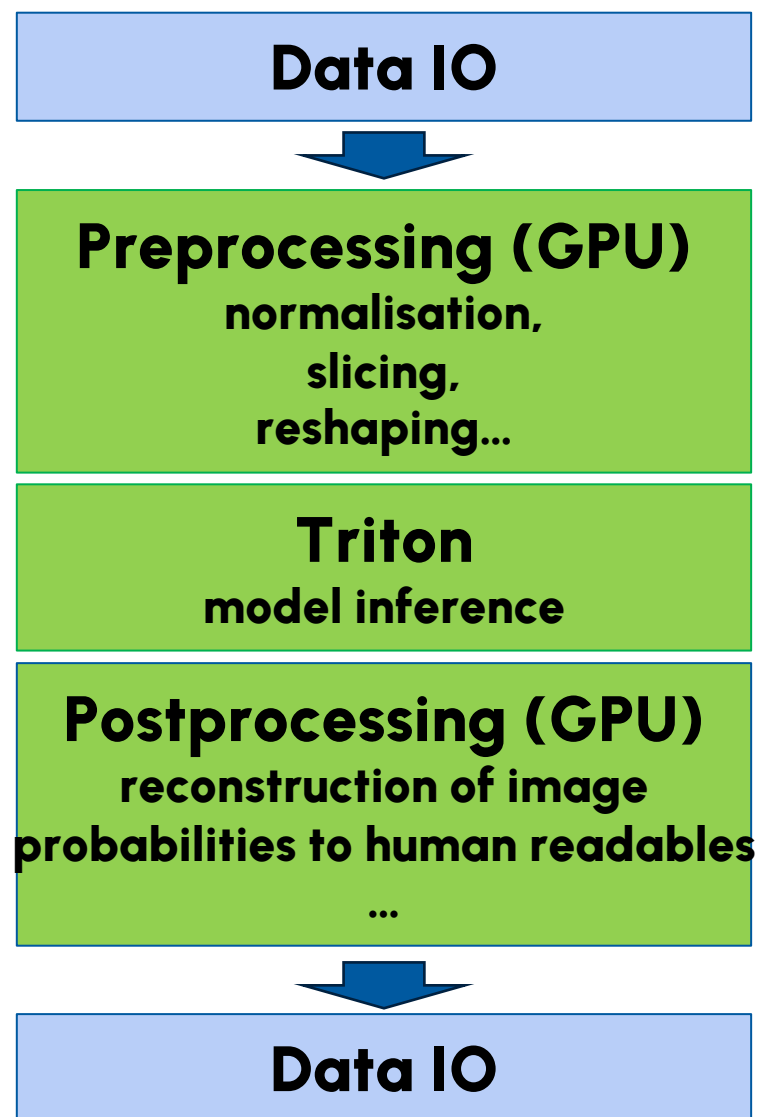
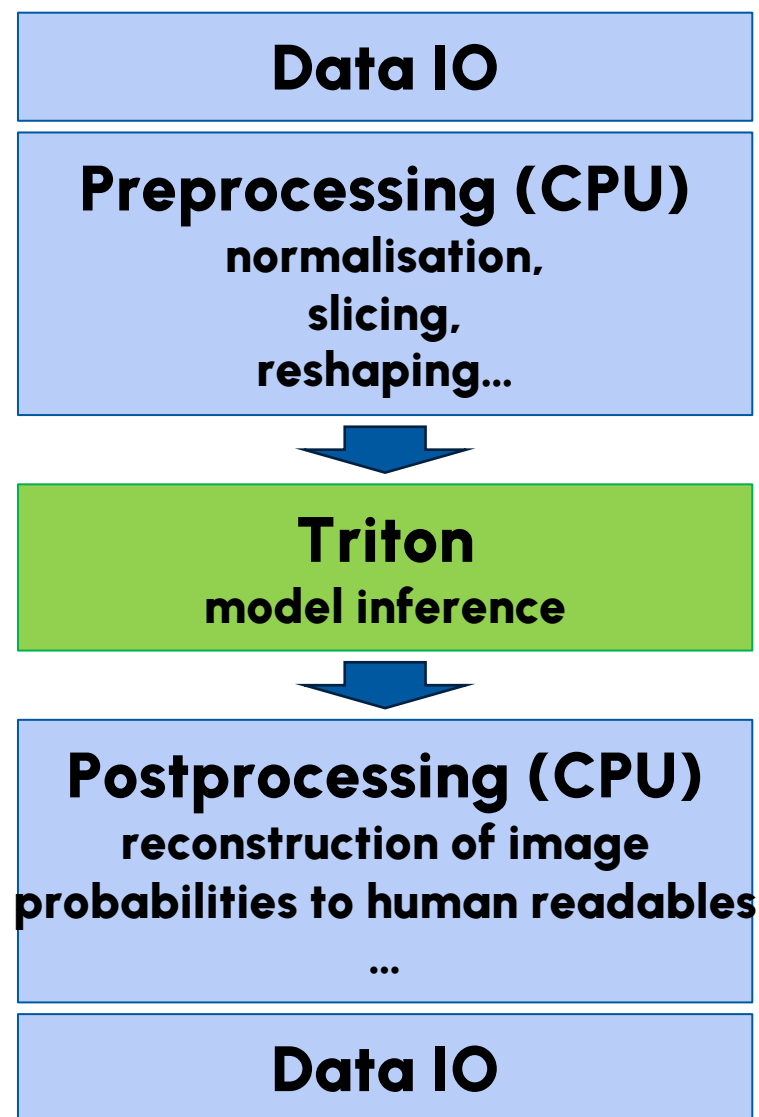
```
name: "Hans_mnist"
platform: "pytorch_libtorch"
max_batch_size : 0
input [
  {
    name: "input__0"
    data_type: TYPE_FP32
    dims: [ 1, 28, 28 ]
    reshape { shape: [ 1, 1, 28, 28 ] }
  }
]
output [
  {
    name: "output__0"
    data_type: TYPE_FP32
    dims: [ 1, 10]
    reshape { shape: [ 1, 10 ] }
  }
]
```

*** a draft for the config.pbtxt can be downloaded from the meta.json form as well**

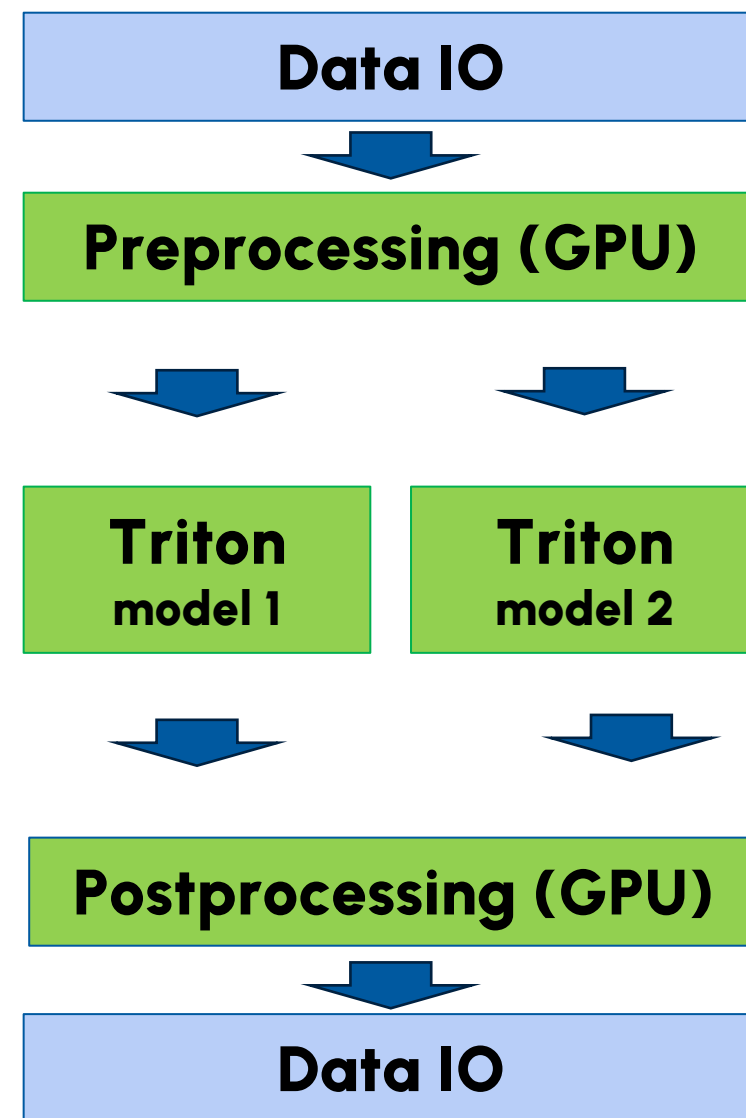
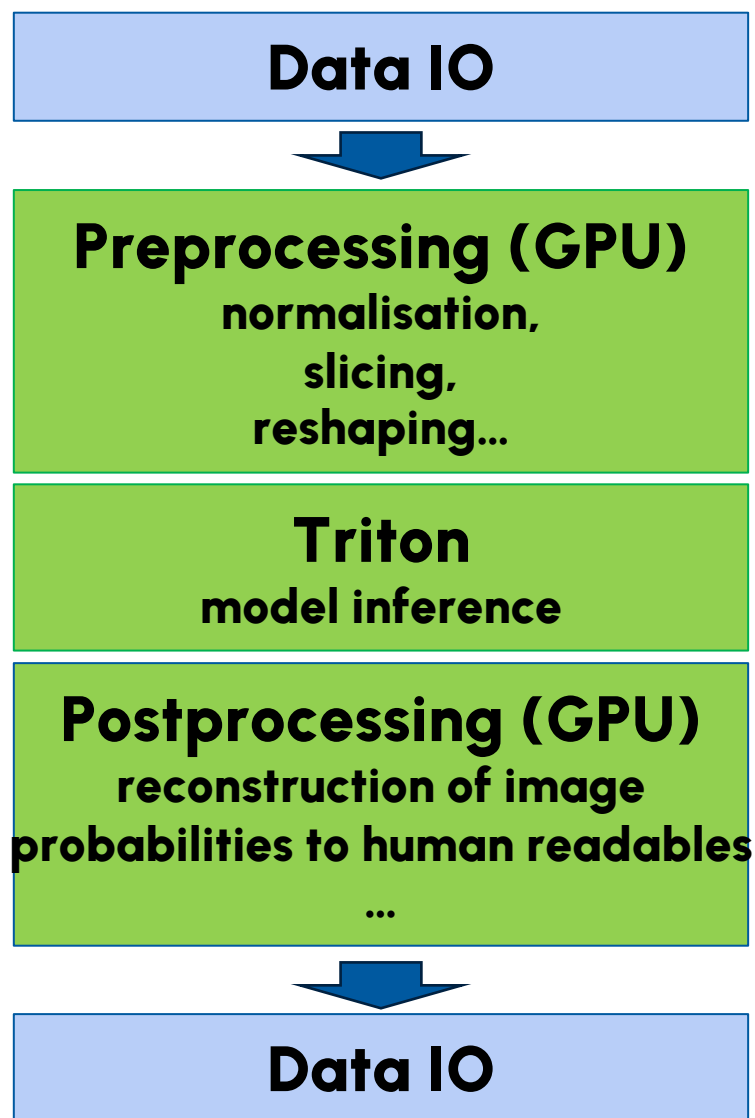
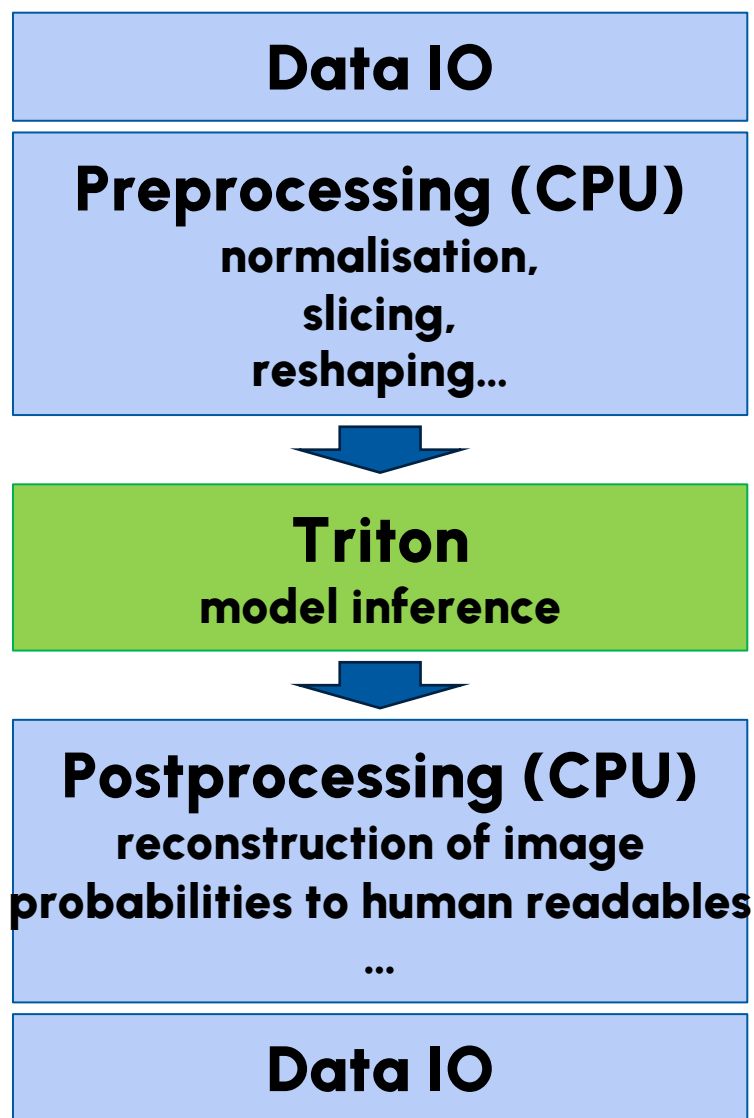
Pre/Post Processing & Triton



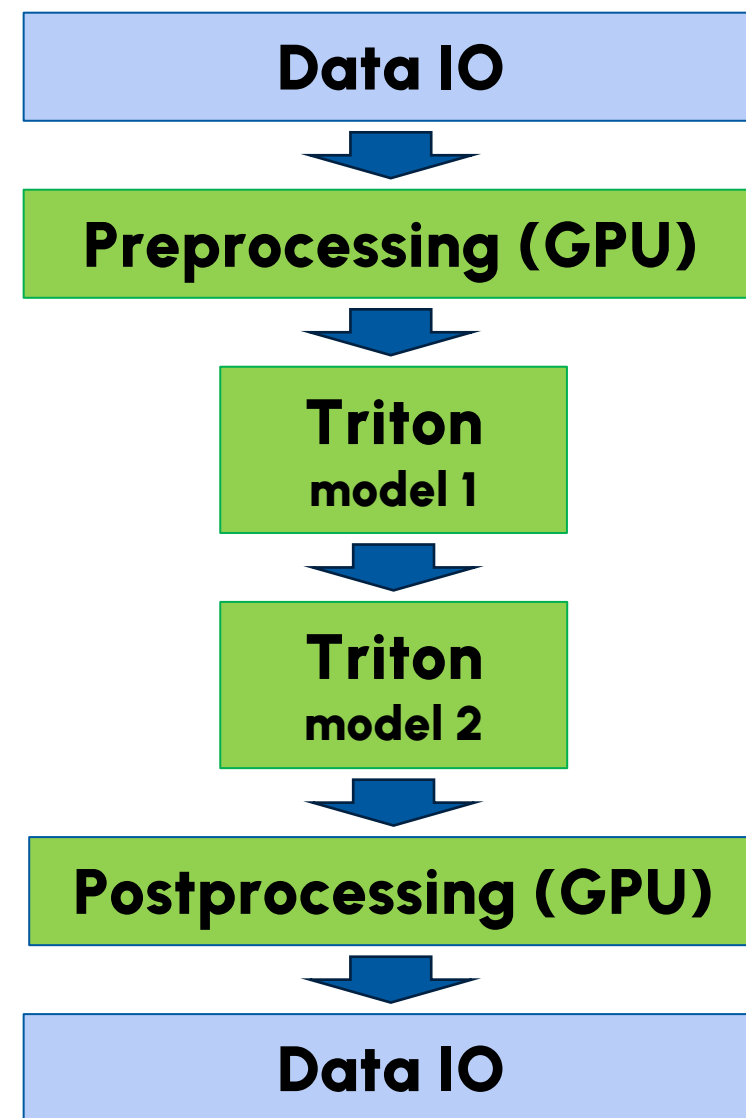
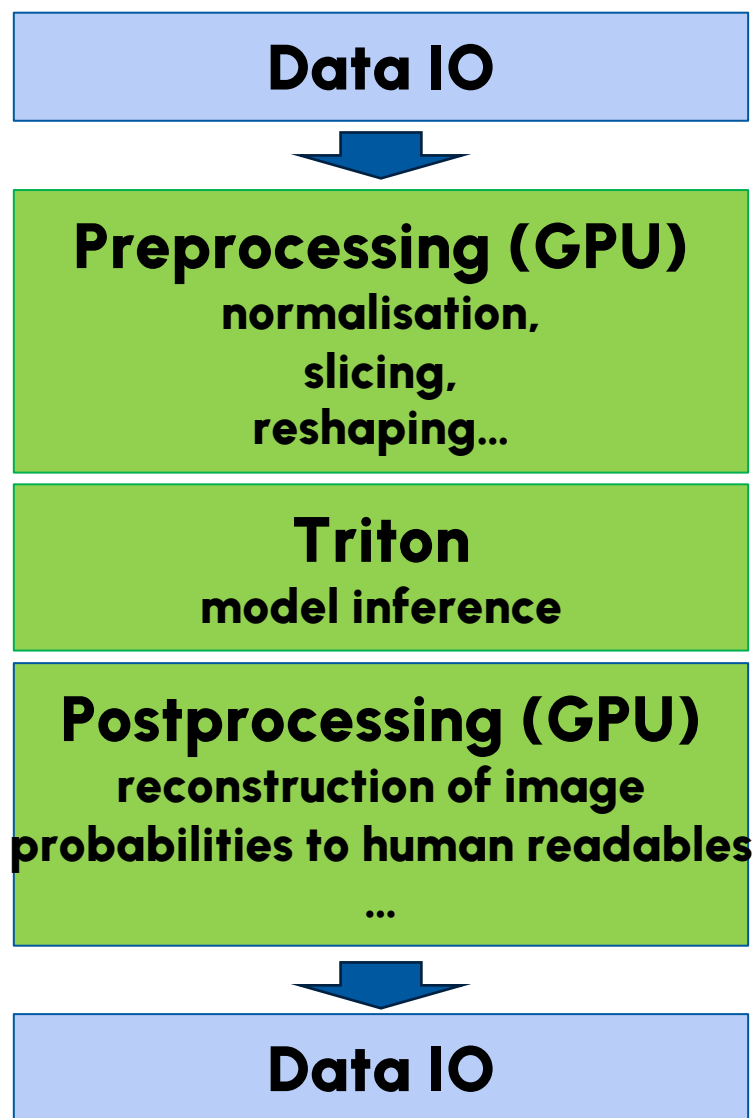
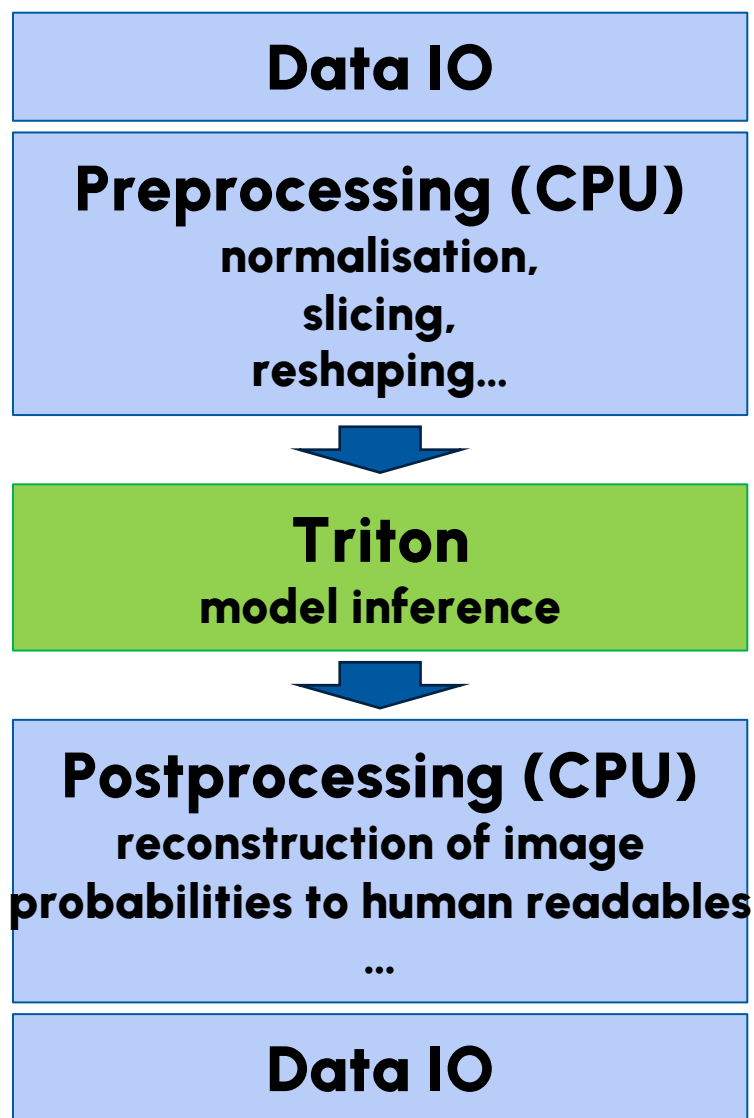
Pre/Post Processing & Triton



Pre/Post Processing & Triton



Pre/Post Processing & Triton



ensemble config.pbtxt

```
name: "ensemble_model"
platform: "ensemble"
input [
  {
    name: "input_volume"
    data_type: TYPE_FP32
    dims: [ 3, -1, -1 ]
  }
]
output [
  {
    name: "output_volume"
    data_type: TYPE_FP32
    dims: [ 3, -1, -1 ]
  }
]
ensemble_scheduling {
  step [
    {
      model_name: "preprocess"
      model_version: 1
      input_map {
        key: "input_volume"
        value: "input_volume"
      }
      output_map {
        key: "output_slices"
        value: "output_slices"
      }
    },
    {
      model_name: "detect_model"
      model_version: 1
      input_map {
        key: "input_slice"
        value: "input_slice"
      }
    }
  ]
}
```

```
ensemble_scheduling {
  step [
    {
      model_name: "preprocess"
      model_version: 1
      input_map {
        key: "input_volume"
        value: "input_volume"
      }
      output_map {
        key: "output_slices"
        value: "output_slices"
      }
    },
    {
      model_name: "detect_model"
      model_version: 1
      input_map {
        key: "input_slice"
        value: "output_slices"
      }
      output_map {
        key: "output_slice"
        value: "processed_slices"
      }
    },
    {
      model_name: "postprocess"
      model_version: 1
      input_map {
        key: "input_slices"
        value: "processed_slices"
      }
      output_map {
        key: "output_volume"
        value: "output_volume"
      }
    }
  ]
}
```



The Model Zoo and you...



- **What is needed to serve your model?**
 - **What are your use-cases?**
 - **Are there models which you urgently need?**
 - **Any features you would like to see?**
-
- **Let's get your first model in the model zoo today?**