Machine learning and deep learning services for the EOSC

The DEEP-Hybrid-DataCloud project

EGI Conference 2020
November 2 – 5, 2020

Álvaro López García
aloga@ifca.unican.es
Spanish National Research Council

EXCELENCIA MARÍA DE MÁEZTU

CSIC

DEEP-HybridDataCloud has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 777435.
DEEP project in 1 slide

- **Designing and Enabling E-Infrastructures for intensive data Processing in a Hybrid DataCloud** (Grant agreement number 777435, Nov 2017 – Apr 2020)

- **Global objective**: Promote the use of intensive computing services by different research communities and areas, and the **support by the corresponding e-Infrastructure** providers and open source projects
  - Focusing on **Machine learning**, **Deep learning**, and **Post processing** services for the European Open Science Cloud

https://deep-hybrid-datacloud.eu
Context: the EOSC

- The **European Open Science Cloud** was launched in 2016 as part of the European Cloud Initiative.
- The EOSC is a **digital platform to support and develop science** and open innovation.
- Europe’s virtual environment for all researchers.
- Providing seamless access to data and interoperable services across borders.
- Access is delivered through the **EOSC portal**.
- Launched in 2018, continuously being enhanced and developed.
DEEP-HybridDataCloud vision

• Provide **specialized cloud services** to develop, exploit and share machine learning and deep learning applications → **Service-Oriented Architectures and platforms**
  – Covering the whole machine learning application development cycle
  – Focused on all types of users and user knowledge

• Transparent **access to specialized computing resources** (accelerators, high performance computing) → **reduce entry barrier**

• Build an **EOSC machine learning marketplace** as an application exchange → **ease of use, foster collaboration, knowledge dissemination**

• **Run the same application everywhere** (laptop, cloud, HPC, etc.) → **ease of use**
The project covers all development phases of a machine learning application:
- Creation, update, improvement of a model (existing or new)
- Model training, test, evaluation
- Model deployment as a service
- Model publication for sharing and reuse
The CRISP-DM cycle

- We are focusing on several phases of the Cross-Industry Standard Process for Data Mining (CRISP-DM) cycle:
  - Modeling, evaluation and deployment phases
- Provide a training and evaluation platform
- Provide a model deployment service
- Provide a catalog to share applications and models
Different users, different needs

- **Category 1**: Deploy a readily trained network for somebody else to use it on his/her data set
  - Domain knowledge

- **Category 2**: Retrain (parts of) a trained network to make use of its inherent knowledge and to solve a new learning task
  - Domain + machine learning knowledge

- **Category 3**: Completely work through the deep learning cycle with data selection, model architecture, training and testing
  - Domain + machine + technological knowledge
Docker containers

- DEEP leverages Docker containers to encapsulate the user application, together with the required environment.
- Containers are automatically built and published to the marketplace.

Docker allows users to run the same application everywhere:
- Using Docker, if available.
- Using uDocker (user space tool, no system-wide install needed) if Docker is not available (like in HPC environments).
  - See: [https://github.com/indigo-dc/udocker/](https://github.com/indigo-dc/udocker/)
- Using serverless components (like Apache OpenWhisk).
The DEEP services

- High level specialized services for exploitation through EOSC
  - DEEP Open Catalog → sharing
  - DEEP training facility → training
  - DEEP as a Service → deployment

- Integration with storage from external initiatives (e.g. eXtreme-DataCloud, EGI Data Hub, etc.)
DEEP Open Catalog

Sharing knowledge, models and applications between users

https://marketplace.deep-hybrid-datacloud.eu
Browsing the marketplace

Explore our marketplace!

Artistic style transfer
- Model | Trainable | Inference | Pre-trained
A module to apply artistic style transfer using PyTorch.

Bird sound classifier
- Model | Trainable | Inference | Pre-trained
Classify audio files among bird species from the Xenocanto dataset.

TF Benchmarks
- Model | Trainable
tf_cnn_benchmarks accessed via DEEPCat API

Object Detection and Classification with PyTorch
- Model | Trainable | Inference | Pre-trained
A trained Region Convolutional/Neural Network (Faster R-CNN) for object detection and classification.

2D semantic segmentation
- Model | Trainable | Inference | Pre-trained
2D semantic segmentation trained on the Vaihingen dataset.

Train an audio classifier
- Model | Trainable | Inference | Pre-trained
Train your own audio classifier with your custom dataset. It comes also pre-trained on the i2i AudioSet classes.
Object Detection and Classification with Pytorch

A trained Region Convolutional Neural Network (Faster R-CNN) for object detection and classification.

Published by DEEP Hybrid DataCloud Consortium
Created: Thu 17 October 2019 - Updated: Mon 15 June 2020

Model Description

This is a plug and play tool for object detection and classification using deep neural networks (Faster R-CNN ResNet 50 FPN Architecture [1]) that were already pretrained on the COCO Dataset. The code uses the Pytorch library, more information can be found at Pytorch-Object-Detection.

The PREDICT method expects an image as input and will return a JSON with the predictions that are greater than the probability threshold. Let's say you have an image of a cat and a dog together and the probability output was 90% a dog and 80% a cat, if you set the threshold to 70%, the only detected object will be the cat, because its probability is greater than 70%.

This module works on uploaded images and gives as output the rectangle coordinates x1,y1 and x2,y2 were the classified object is located. It also provides you the probability of the classified detected object.
Local inference and prediction

(udocker) alvaro:~ $ udocker pull deephdc/deep-oc-obj_detect_pytorch
Info: downloading layer sha256:2746e4a2851c9e18bf7ff0429c18fd7522acc14fa4c7ec8ab37ba5ebeadbc584
...

(udocker) alvaro:~ $ udocker create deephdc/deep-oc-obj_detect_pytorch
(udocker) alvaro:~ $ udocker run -p 5000:5000 deephdc/deep-oc-obj_detect_pytorch
Warning: this container exposes TCP/IP ports
Warning: non-existing user will be created

*****************************************************************************
*                                                                            *
*               STARTING 9bb8179d-c731-348c-bbf4-e43045ee8936                *
*                                                                            *
*****************************************************************************
executing: deepaas-run

Welcome to the DEEPaaS API API endpoint. You can directly browse to the API documentation endpoint to check the API using the builtin Swagger UI or you can use any of our endpoints.

API documentation: http://0.0.0.0:5000/ui
API specification: http://0.0.0.0:5000/swagger.json
V2 endpoint: http://0.0.0.0:5000/v2

-------------------------------------------------------------------------
Local inference and prediction

Welcome to the DEEPaaS API API endpoint. You can directly browse to the API documentation endpoint to check the API using the builtin Swagger UI or you can use any of our endpoints.

API documentation: http://0.0.0.0:5000/ui
API specification: http://0.0.0.0:5000/swagger.json
V2 endpoint: http://0.0.0.0:5000/v2

DEEP as a Service API endpoint 1.1.0

This is a REST API that is focused on providing access to machine learning models. By using the DEEPaaS API users can easily run a REST API in front of their model, thus accessing its functionality via HTTP calls.

Currently you are browsing the Swagger UI for this API, a tool that allows you to visualize and interact with the API and the underlying model.

- Project website
- Project documentation
- Model marketplace

API documentation

versions

GET  /v2/  Get V2 API version information

GET  /  Get available API versions

def

GET  /v2/debug/  Return debug information if enabled by API.

models

GET  /v2/models/  Return loaded models and its information

GET  /v2/models/obj_detect_pytorch/  Return model’s metadata

Welcome to the DEEPaaS API API endpoint. You can directly browse to the API documentation endpoint to check the API using the builtin Swagger UI or you can use any of our endpoints.

API documentation: http://0.0.0.0:5000/ui
API specification: http://0.0.0.0:5000/swagger.json
V2 endpoint: http://0.0.0.0:5000/v2
Local inference and prediction

Welcome to the DEEPaaS API API endpoint. You can directly browse to the API documentation endpoint to check the API using the builtin Swagger UI or you can use any of our endpoints.

API documentation: http://0.0.0.0:5000/ui
API specification: http://0.0.0.0:5000/swagger.json

V2 endpoint: http://0.0.0.0:5000/v2

Hybrid-DataCloud

Welcome to the DEEPaaS API API endpoint. You can directly browse to the API documentation endpoint to check the API using the builtin Swagger UI or you can use any of our endpoints.

API documentation: http://0.0.0.0:5000/ui
API specification: http://0.0.0.0:5000/swagger.json

V2 endpoint: http://0.0.0.0:5000/v2

Local inference and prediction
DEEPaaS: DEEP as a Service

Automatic deployment of catalog modules to exploit their functionality

https://deepaas.deep-hybrid-datacloud.eu
DEEPaaS: automated deployment

DEEP as a Service

Welcome to the DEEP as a Service production endpoint.

Available modules and endpoints

You can access the list of deployed modules in any of the following formats:

- HTML
- JSON
- TXT

Swagger interfaces

- Swagger UI for "image-classification-tf-dicom" | Marketplace entry
- Swagger UI for "dogs_breed_det" | Marketplace entry
- Swagger UI for "senseg_vahingen" | Marketplace entry
- Swagger UI for "5df5d6e845aebd4aebc825c8" | Marketplace entry
Object Detection and Classification with Pytorch

A trained Region Convolutional Neural Network (Faster R-CNN) for object detection and classification.

Published by DEEP-Hybrid DataCloud Consortium
Created: Thu 17 October 2019 - Updated: Mon 15 June 2020

Model Description

This is a plug-and-play tool for object detection and classification using deep neural networks (Faster R-CNN ResNet50 FPN Architecture [1]) that were already pretrained on the COCO Dataset. The code uses the Pytorch library, more information can be found at Pytorch-Object-Detection.

The PREDICT method expects an image as input and will return a JSON with the predictions that are greater than the probability threshold. Let’s say you have an image of a cat and a dog together and the probability output was 60% a dog and 80% a cat, if you set the threshold to 70%, the only detected object will be the cat, because its probability is greater than 70%.

This module works on uploaded images and gives as output the rectangle coordinates x1,y1 and x2,y2 where the classified object is located. It also provides you the probability of the classified detected object.
Marketplace entry
DEEP training facility

1-click development and training environments in Cloud and HPC resources

https://train.deep-hybrid-datacloud.eu
DEEP training dashboard

DEEP Development Environment

The DEEP Development Environment provides a ready to use JupyterLab instance that enables you to develop code using Jupyter notebooks, text editors, terminals, and custom components in a flexible, integrated, and extensible manner.

Explore the Marketplace

- Dogs breed detector
  - Identify a dogs breed on the image (133 known breeds)
  - Train module
  - Marketplace

- DEEP OC Massive Online Data Streams
  - Deep learning for proactive network monitoring and security protection
  - Train module
  - Marketplace

- DEEP OC Retinopathy Test
  - A Tensorflow model to classify Retinopathy
  - Train module
  - Marketplace

- Conus species classifier
  - Classify conus images among 70 species
  - Train module
  - Marketplace

- Plants species classifier
  - Classify plant images among 10k species from the PlantNet database
  - Train module
  - Marketplace

- Train an image classifier
  - Train your own image classifier with your custom dataset. It comes also pre-trained on the 1K ImageNet classes.
  - Train module
  - Marketplace
DEEP development environment

https://youtu.be/J_I_xWiBGNA
DEEP training facility

1. Select a module from the Open Catalogue

2. Choose your configuration parameters

3. View your deployments

4. Monitor the training history

https://deep-hybrid-datacloud.eu
DEEPaaS API: models as services

Exposing models through a common and standard-based API

https://deepaas.readthedocs.io
The project covers all development phases of a machine learning application:

- Creation, update, improvement of a model (existing or new) → DEEP Open Catalog and DEEP training facility
- Model training, test, evaluation → DEEP training facility
- Model deployment as a service → DEEP as a Service
- Model publication for sharing and reuse → DEEP Open Catalog
Second software release and platform released January 2020

- Full support for the whole machine learning application life cycle
  - Sharing and reuse through DEEP Open Catalogue
  - Training, testing and evaluation on DEEP training facility
  - Live deployment on DEEPaaS
- Transparent access to Cloud and HPC resources
Conclusions

- DEEP provides a comprehensive, service-oriented, framework to develop machine learning, deep learning and artificial intelligence applications
  - Exploiting distributed e-Infrastructure and HPC resources, with special focus on the EOSC
  - Covering the whole development phases (CRISP-DM and machine learning lifecycle)
  - High level and added value services that can be used standalone or as a whole platform solution

- Services at production level with TRL>=8

- Industry opportunities: engagement of industrial stakeholders through collaboration with EOSC-Digital Innovation Hub
Request access through EOSC portal

DEEP Services
Training facility: https://train.deep-hybrid-datacloud.eu/
DEEPaaS: https://deepaas.deep-hybrid-datacloud.eu/

Documentation
http://docs.deep-hybrid-datacloud.eu

DEEP videos
https://www.youtube.com/playlist?list=PLJ9x9Zk1O-J_UZfNO2uWp2pFMmbwLvzXa
Contact
aloga@ifca.unican.es, deep-info@listas.csic.es

Web page
https://deep-hybrid-datacloud.eu/the-platform/

Social media
https://twitter.com/DEEP_eu
https://twitter.com/IFCA_Computing
Thank you
Any Questions?

https://deep-hybrid-datacloud.eu

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 777435.