Applied Data Science Project

L12 – Colaboratory

Giuseppe Rizzo

Turin, October 10, 2022

Pillars

Design

Manage

Develop

Communicate

Development

It is where the magic happens

An artificial intelligence system is created to generate the outputs that meet objectives and requirements with the involvement of a team tasked on activities with due dates

Objectives and requirements have been defined in the Design pillar

Activities and due dates in the Manage pillar

Knowledge tools

Development environment

Foundation models
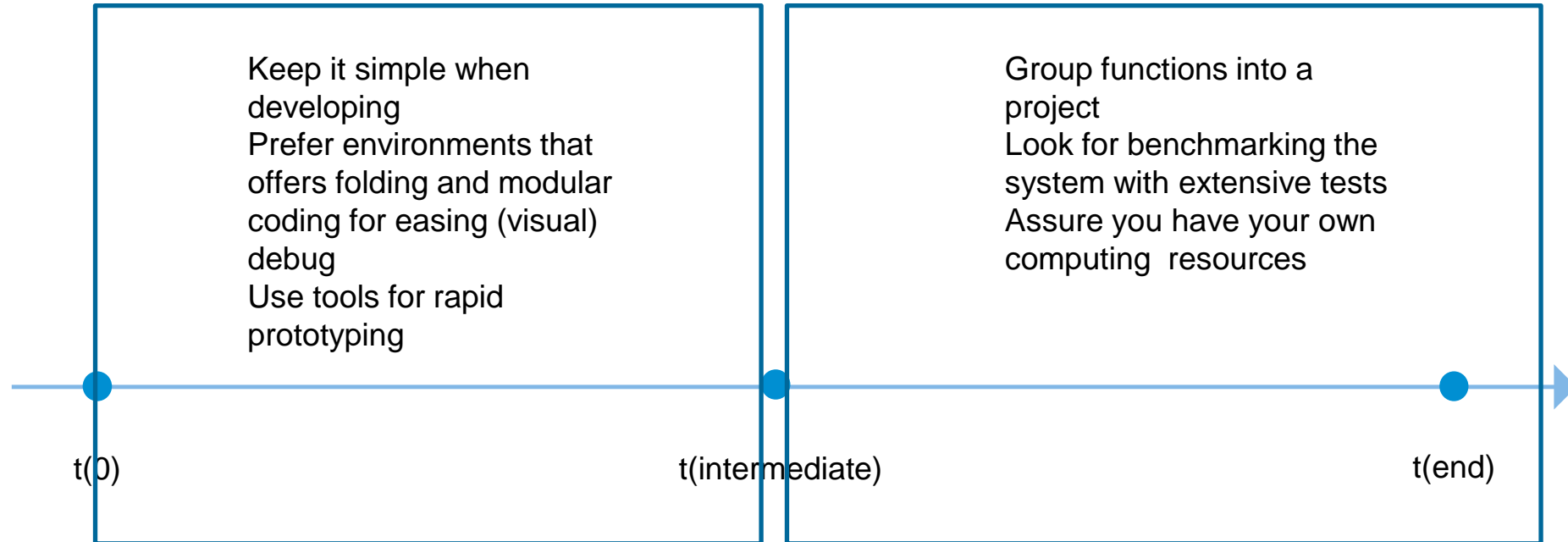
Version control

Internal communication (with team mates)

Development

- "Divide and rule"

- Tasks are mapped into modules

- One module has one lead developer and, eventually, contributors

Python is considered the default language for developing machine intelligence nowadays

- Choose the programming language according to:
  - ecosystem of software modules that you can utilize
  - easiness of model integration
  - familiarity. Do not be afraid to switch to another (similar programming language) since they share most of the features and development patterns

Development timeline

Keep it simple when developing
Prefer environments that offers folding and modular coding for easing (visual) debug
Use tools for rapid prototyping

Group functions into a project
Look for benchmarking the system with extensive tests
Assure you have your own computing resources

t(0)

t(intermediate)

t(end)

Colab Notebook
Focus of the course

Hubs for coding (MLHub) or in IDE (VS Code)

Colaboratory for rapid prototyping

https://colab.research.google.com

An application where to develop, share, and also test on dedicated hardware (GPU to speed up the computing)

Overview

add a block for writing explanations

Share

share with collaborators

add a block for code writing

Welcome To Colaboratory

File   Edit   View   Insert   Runtime   Tools   Help

+ Code   + Text   Copy to Drive   Connect ▾   Editing

Table of contents of the notebook

What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your browser, with

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!

Getting started

The document you are reading is not a static web page, but an interactive environment ca a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores i a variable, and prints the result:

```
[ ]   seconds_in_a_day = 24 * 60 * 60
      seconds_in_a_day
```

```
      86400
```

To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut "Command/Ctrl+Enter". To edi code, just click the cell and start editing.

Variables that you define in one cell can later be used in other cells:

```
[ ]   seconds_in_a_week = 7 * seconds_in_a_day
      seconds_in_a_week
```

Notebook containing both code blocks and explanations in a narrative fashion

Menu bar

Welcome To Colaboratory

File   Edit   View   Insert   **Runtime**   Tools   Help

define the runtime configurations, among those the hardware acceleration if needed

Notebook settings

Hardware accelerator

GPU

To get the most out of Colab, avoid using a GPU unless you need one. Learn more

☐ Omit code cell output when saving this notebook

Cancel   Save

Share

Connect your Google Drive

Enable authorization to import data from Google Drive

```
1    from google.colab import drive

2    drive.mount('/content/drive/')
```

List the files in your drive

```
1    !ls "/content/drive/My Drive/"
```

Running with Google Colab

Upload YOUR_PYTHON_FILE.py to Google Drive & Run with Google Colab

```
1    !python3 "/content/drive/My Drive/Colab
     Notebooks/YOUR_PYTHON_FILE.py"
```

Run with Google Colab to Download YOUR_PYTHON_FILE.py from Google Drive

```
1    from google.colab import files

2    files.download('/content/drive/My Drive/Colab
     Notebooks/YOUR_PYTHON_FILE.py')
```

Bash commands

Bash commands are executed with the environment "!"

Download an external file

```
1    !wget
     http://ai.stanford.edu/~amaas/data/sentiment/aclIm
     db_v1.tar.gz -P "/content/drive/My Drive/Colab
     Notebooks"
```

Clone a repository

```
1    !git clone https://github.com/pytorch/examples.git
```

Colab == virtual environment

The environment can be customized with the addition of python packages

Install

```
1    !pip install -q datasets
```

Show a version

```
1    !pip show datasets
```

Example

+ Code + Text

Note: make sure that Runtime -> Change runtime type -> hardware accellerator -> GPU is selected

Install the **libraries** used by this Colab notebook in the virtual environment

```
[4]  !pip install -q datasets
     !pip install -q transformers
     !pip install -q simpletransformers
```

Load IMDb review dataset

https://www.imdb.com/interfaces/

```
[5]  import pandas as pd
     from datasets import load_dataset

     dataset_train = load_dataset('imdb',split='train')
     dataset_train.rename_column('label', 'labels')
     train_df=pd.DataFrame(dataset_train)

     dataset_test = load_dataset('imdb',split='test')
     dataset_test.rename_column('label', 'labels')
     test_df=pd.DataFrame(dataset_test)
```

```
Downloading builder script: 100%  4.31k/4.31k [00:00<00:00, 42.2kB/s]
Downloading metadata: 100%  2.17k/2.17k [00:00<00:00, 26.1kB/s]
Downloading and preparing dataset imdb/plain_text (download: 80.23 MiB, generated: 127.02 MiB, post-pr
Downloading data: 100%  84.1M/84.1M [00:11<00:00, 15.6MB/s]
Dataset imdb downloaded and prepared to /root/.cache/huggingface/datasets/imdb/plain_text/1.0.0/2fdd8b
WARNING:datasets.builder:Found cached dataset imdb (/root/.cache/huggingface/datasets/imdb/plain_text/
```

TORINO

15

Thank you for your attention.

Questions?

CONTACTS

Giuseppe Rizzo

Program Manager (LINKS Foundation) and
Adjunct Professor (Politecnico di Torino)

giuseppe.rizzo@polito.it