[DM 2024] Lab 1 Environment settings

Hi everybody,

We will have our first lab session on September 23 (Monday) 9:00 AM on our Youtube Stream / Classroom. Please be on time.

We highly recommend you to attend the session with your personal laptop (that way you'll also have your environment set for the homework). These are some instructions for you to set up the environment:

1.A. Install Python:

We are using Python 3.9.6. You can use newer version, but use at your own risk. Follow the installation instructions below:

https://www.python.org/downloads/release/python-396/

Once you install the python, you can check whether your python is ready by opening a "terminal" windows (Linux/MacOS) or a "Command Prompt" window.

```
Command Prompt
                        ×
Microsoft Windows [Version 10.0.22631.4112]
 (c) Microsoft Corporation. All rights reserved.
C:\Users\didif>python --version
Python 3.9.6
```

1.B. Set up a Virtual Environment:

If you use Python a lot for other projects, you might need to isolate different installation packages for compatibility purposes. If this is the first time you use Python or if you don't encounter any compatibility issues, you can skip this step.

To have virtual environments, you need to install Anaconda or virtualenv. Here are the instructions for how to install and use each one:

https://www.anaconda.com/distribution/

https://virtualenv.pypa.io/en/latest/installation.html

2. Install libraries:

We will use the following Python libraries for the lab: Jupyter Notebook, Scikit learn, Numpy, Pandas, NLTK, Matplotlib, Plotly and PAMI.

Once you have installed Python 3 (and optionally Anaconda), open a "terminal" windows (Linux/MacOS) or a "Command Prompt" window and type the following commands followed by "Enter":

pip3 install jupyter pip3 install numpy pip3 install pandas pip3 install matplotlib pip3 install plotly pip3 install nltk pip3 install scikit-learn pip3 install seaborn pip3 install pami

pip3 install umap-learn

Open a "Terminal" window (Linux/MacOs) or a "Command Prompt" window (windows) and type the following line (followed by Enter): python (or python3) Then type the following lines (followed by Enter): import nltk nltk.download('punkt')

Note: you might need to add "Sudo" at the beginning if you use Linux, and you will be asked to input your password.

🔒 IDLE Shell 3.9.6 File Edit Shell Debug Options Window Help D64)] on win32

Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (I Type "help", "copyright", "credits" or "license()" for more information. >>> import nltk >>> nltk.download('punkt') [nltk data] Downloading package punkt to [nltk data] C:\Users\didif\AppData\Roaming\nltk data...

[nltk_data] Unzipping tokenizers\punkt.zip. True 3. Run Jupyter Python and check your environment:

Open a new Jupyter notebook server. In order to do this, open a "terminal" windows (Linux/MacOS) or a "Command Prompt" window and type the following commands followed by "Enter":

jupyter notebook

If you receive an error message, zsh: command not found: jupyter, type the following commands instead.

python3 -m notebook

Just like the image below:

C:\Users\didif>jupyter notebook Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extension https://jupyter-notebook.readthedocs.io/en/latest/migrate_to_notebook7.html Please note that updating to Notebook 7 might break some of your extensions. [W 09:25:24.626 NotebookApp] Loading JupyterLab as a classic notebook (v6) extension. [I 2024-09-02 09:25:24.632 LabApp] JupyterLab extension loaded from C:\Users\didif\anaconda3\Lib\site-packages\jupyterla [I 2024-09-02 09:25:24.632 LabApp] JupyterLab application directory is C:\Users\didif\anaconda3\share\jupyter\lab [I 09:25:26.078 NotebookApp] The port 8888 is already in use, trying another port. [I 09:25:26.079 NotebookApp] The port 8889 is already in use, trying another port. [I 09:25:26.080 NotebookApp] Serving notebooks from local directory: C:\Users\didif [I 09:25:26.080 NotebookApp] Jupyter Notebook 6.5.4 is running at: [I 09:25:26.080 NotebookApp] http://localhost:8890/?token=16c164cbeab4c8465e1700fbd68d02f6791d7086589374dc [I 09:25:26.080 NotebookApp] or http://127.0.0.1:8890/?token=16c164cbeab4c8465e1700fbd68d02f6791d7086589374dc [I 09:25:26.080 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation). [C 09:25:26.122 NotebookApp]

A window like the one below should open in your browser. Please go to the "New" button on the top right cornerand select "Python 3".



https://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/Running%20Code.html https://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/Notebook%20Basics.html

This will open a new notebook. You will be able to run "Cells" of code and get the outputs printed below, as well as cells of text. If you want to learn more on how to use a notebook, read the documentation below:

Once you opened a new notebook, please paste the script below in a cell and press the "Run" Button (or the "Shift" + "Enter" keys). Make sure you have no errors!

from sklearn.feature_extraction.text import CountVectorizer

In [1]: # import library

import pandas as pd import numpy as np import nltk from sklearn.datasets import fetch_20newsgroups

import plotly as py import math import PAMI import umap %matplotlib inline # prepare dataset categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med'] twenty_train = fetch_20newsgroups(subset='train', categories=categories, shuffle=True, random_state=42) It should look like this: DM2024-Lab1-Announcement Last Checkpoint: a few seconds ago (autosaved) Widgets Python 3 (ipykernel) O Help Trusted View Insert Cell Kernel ► Run ■ C → Markdown errors! In [1]: # import library import pandas as pd import numpy as np import nltk from sklearn.datasets import fetch_20newsgroups from sklearn.feature_extraction.text import CountVectorizer import plotly as py import math import PAMI import umap %matplotlib inline # prepare dataset categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med'] twenty train = fetch 20newsgroups(subset='train', categories=categories, shuffle=True, random state=42)

1. Sign up on Github: https://github.com/

4. Create a GIT account:

2. Install Git. To install git in windows, you'll need to download it from here:

GIT is a version control system that tracks changes to code. We won't be using it much during the tutorial, but you will have to hand out your code through this platform. To set up Git, you'll need to:

https://gitforwindows.org/

In linux/MacOS, open a "Terminal" window and run the following command: sudo apt install git-all

You can check if you have git installed in your computer by typing: git --version

C:\Users\didif>git --version

3. Add your "credentials" to git. Open a "Terminal" window and run the following commands(please use the same email you used to register for your Git account): git config --global user.name "YOUR USERNAME" git config --global user.email "your_email@gmail.com"

It should look like this: Command Prompt

(c) Microsoft Corporation. All rights reserved.

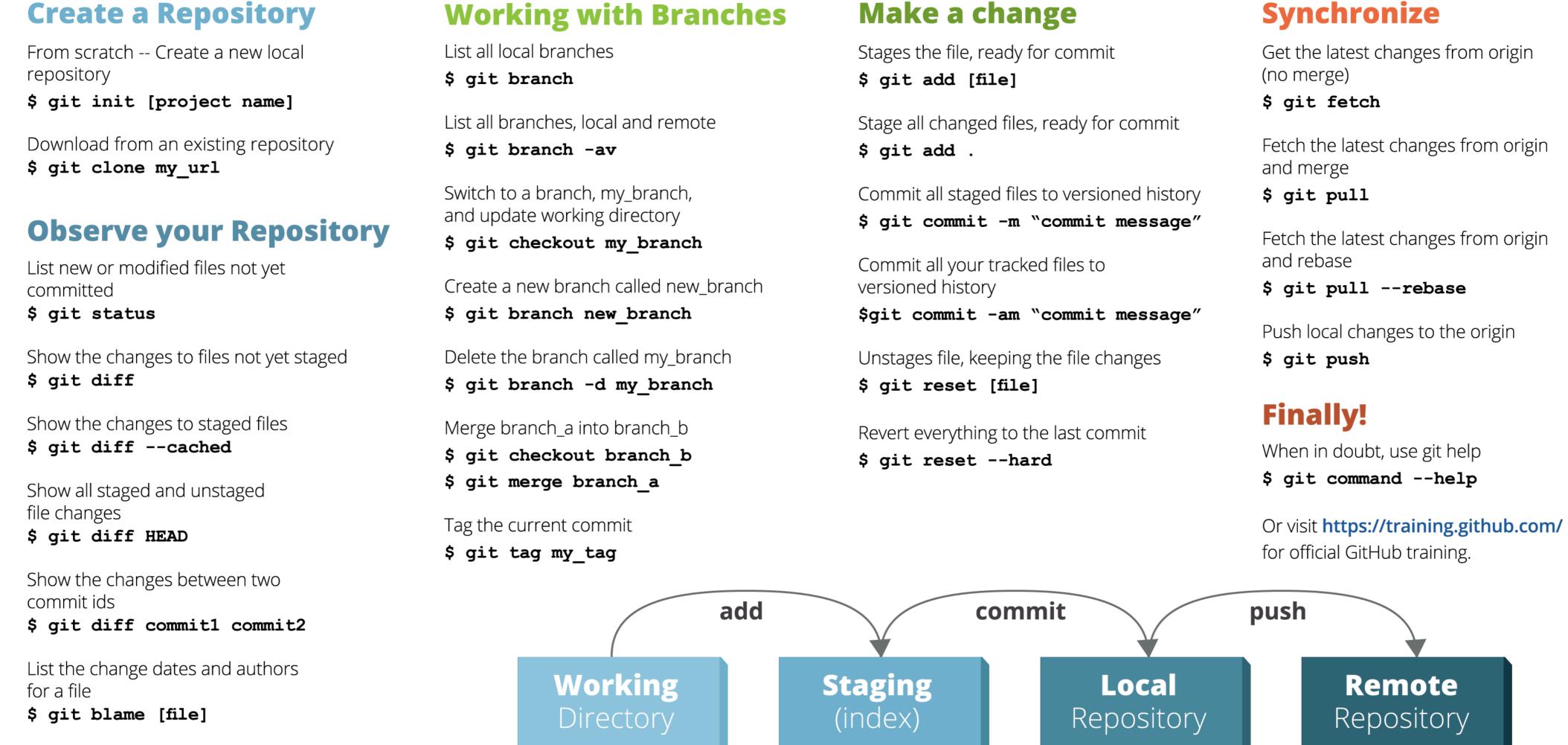
Microsoft Windows [Version 10.0.22631.4112]

git version 2.37.3.windows.1 C:\Users\didif>git config --global user.name "didiersalazar" C:\Users\didif>git config --global user.email "113065892@office365.nthu.edu.tw"

Note: If you're not comfortable with the command line, you may also download the Sourcetree application: https://desktop.github.com/ Here's a cheatsheet of commands(for those who won't use the command line and not the app):

Git Cheat Sheet

Create a Repository Working with Branches



Show the file changes for a commit id and/or file \$ git show [commit]:[file] fetch reset Show full change history \$ git log reset [commit] Show change history for file/directory pull including diffs \$ git log -p [file/directory] If you have issues to install/set up Python 3+ environment in your computer, please try using Kaggle kernels. In order to use a Kaggle Kernel: 1. You need to create an account on kaggle (https://www.kaggle.com/).

4. In order to check everything is working, try our Test script code (section 3) in the Notebook and run the code in a block by pressing shift + enter. Note that for the PAMI library because it is an external library that Kaggle does not have installed by

Remote

Repository

© Save Version

€5

No input attached

Attach a Kaggle dataset, model, or competition

Output (68KiB / 19.5GiB)

/kaggle/working

!pip install pami If all the settings are correct, it will look like this:

*

<>

The TAs

Note:

Please Run the script above in step 3 Please make sure the computer you will use during the lab session can work before the lab! notebook516d0634... Draft saved

 Draft Session (4m)

D + Notebook Draft Session Ø # import library Input No Accelerator import pandas as pd import numpy as np Disk Session + Add Input 2.4_{GIB} 4m from sklearn.datasets import fetch_20newsgroups Ш Max 57.6GiB 12 hours

CPU

CPU

0.00%

RAM

871.6мів

Max 30GiB

!pip install pami import PAMI 目 %matplotlib inline # prepare dataset Collecting pami Downloading pami-2024.8.28-py3-none-any.whl.metadata (78 kB)

from sklearn.feature_extraction.text import CountVectorizer

2. Go to the Kernels page (https://www.kaggle.com/kernels), click "New Kernel" and choose "Notebook".

3. Open the session metrics in the Settings (right side).

default, you need to add the following line of code for it to be installed:

Run Settings Add-ons Help

import plotly as py

import math

import umap

categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med'] twenty_train = fetch_20newsgroups(subset='train', categories=categories, shuffle=True, random_state=42) ----- 78.1/78.1 kB 2.8 MB/s eta 0:00:00 Requirement already satisfied: psutil in /opt/conda/lib/python3.10/site-packages (from pami) (5.9.3) Requirement already satisfied: pandas in /opt/conda/lib/python3.10/site-packages (from pami) (2.2.2)

Collecting resource (from pami) Downloading Resource-0.2.1-py2.py3-none-any.whl.metadata (478 bytes)

Requirement already satisfied: plotly in /opt/conda/lib/python3.10/site-packages (from pami) (5.22.0) Requirement already satisfied: matplotlib in /opt/conda/lib/python3.10/site-packages (from pami) (3.7.5) Collecting validators (from pami) Downloading validators-0.34.0-py3-none-any.whl.metadata (3.8 kB)

Requirement already satisfied: Pillow in /opt/conda/lib/python3.10/site-packages (from pami) (9.5.0) Requirement already satisfied: numpy in /opt/conda/lib/python3.10/site-packages (from pami) (1.26.4) Collecting sphinx (from pami) Downloading sphinx-8.0.2-py3-none-any.whl.metadata (6.2 kB) Collecting discord.py (from pami) Downloading discord.py-2.4.0-py3-none-any.whl.metadata (6.9 kB) Requirement already satisfied: networkx in /opt/conda/lib/python3.10/site-packages (from pami) (3.3)

Requirement already satisfied: urllib3 in /opt/conda/lib/python3.10/site-packages (from pami) (1.26.18) Table of contents Requirement already satisfied: sphinx-rtd-theme in /opt/conda/lib/python3.10/site-packages (from pami) (0.2.4) Requirement already satisfied: deprecated in /opt/conda/lib/python3.10/site-packages (from pami) (1.2.14) Requirement already satisfied: wrapt<2,>=1.10 in /opt/conda/lib/python3.10/site-packages (from deprecated->pami) (1.16.0)

Good luck and see you on Monday!

>_ Important Note: If you're having installation issues with all of this, please ask your classmates or TAs for help well ahead of the lab session. Best regards,