Google Colab 101

11785-Introduction to Deep Learning, Carnegie Mellon University

Author: Haoxuan Zhu (haoxuanz@andrew.cmu.edu)

Spring 2021

Brief Intro

Colaboratory is a notebook that allows us to write and execute Python code through the browser. You can consider it as an online version of Jupyter Notebook (it's ok if you do not know what it is. You just need to know that this environment requires no setup and runs entirely in the cloud.) It provides "free" access to GPUs/TPUs under restrictions. You can also pay for **Colab Pro** for much more powerful GPUs.

I have also prepared you a Colab Notebook to help you practice. You **SHOULD** test yourself here after finishing this tutorial. But first, let's clear some grounds and learn/review some concepts! :)

How does Colab work?

Colab lets your code run on *Virtual Machine(s)*. Actually, the entire Colab runs in a cloud VM. If you execute

```
!cat /etc/*release
```

you will find out that the notebook is running on top of `Ubuntu`.

Whenever you click `Connect`, Colab will allocate an instance for you. An instance is a virtual machine (VM) hosted on Google's infrastructure. Per your request, it can be a CPU, GPU, or TPU instance. Hence, when you upload/download files from Colab, you are actually accessing the remote storage of the instance (not your Google Drive). Note that all files will be gone once you disconnect, as the whole VM will be cleared.

What does Colab provide?

Almost everything you need to compile and run Python code. In most cases, you do not need to set up the environment. Once you are connected to an instance, you can directly write code in the text boxes, and run them by `shift+enter`. Even better, Colab has pre-installed most packages, modules ,and libraries for us. You can check via

```
!pip list
```

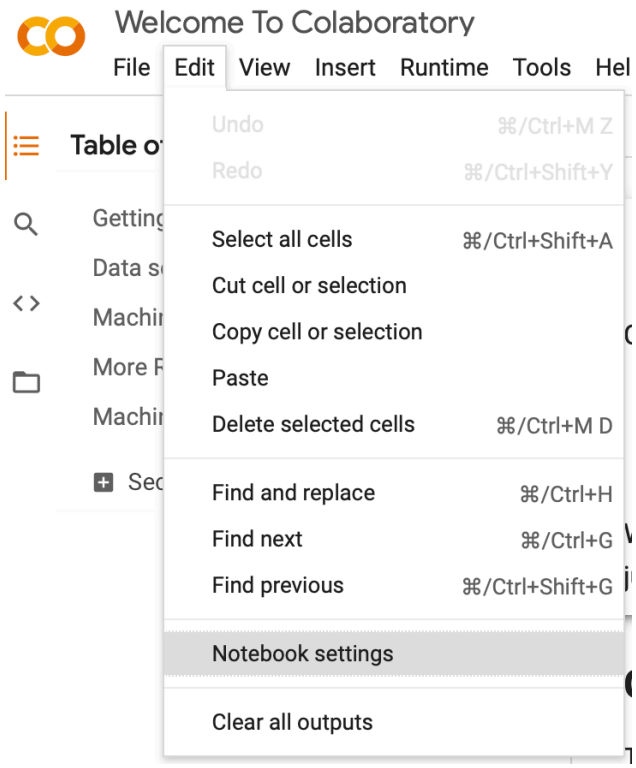
Some Python packages may not be preinstalled on Google Colab. For those you may use `!pip install <package_name>` to install them yourself.

If you still have questions about Colab, you can check its official FAQ page and the official quick tutorial.

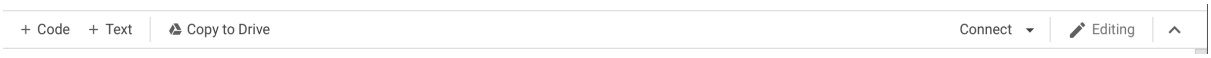
Colab Settings

Google Colab Connection

1. You can connect to Google Colab via your google account by going to https://colab.research.google.com/. Click on `New Notebook` and start working!
2. Once you are in the new notebook, you need to choose what Processor you want. So click on `Edit-> Notebook Settings` and select a GPU when you need one.



3. You will see a `Connect` button on top and tada...you have a free GPU now!



Mounting your Google drive

1. Once you are connected to a notebook, there will be times when you would want to connect the storage of this notebook to your drive for various reasons, like, getting input data, saving or loading your models, etc.
2. To do that, type the below command and authenticate as instructed.

```
from google.colab import drive
drive.mount('/content/gdrive')
```

Then, follow the instruction and enter your authorization code

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6qk8qdg
Enter your authorization code:
```

You are good to go if you see the following message

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

If you want to navigate inside any folder to fetch data/save models, you can do `cd` or `oschdir` to it.

```
%cd gdrive/MyDrive/
```

tips: you can run bash commands directly from Colab by adding an `!` before your `cmd`. However, if you use `!cd gdrive`, you will not enter the `gdrive` folder. You are recommended to **always** check where you are with `!pwd`.

Saving/Loading Models to/from Google Drive

An important part of the models you are building is that you save them from time to time and then reload them later to improvise. A simple way to do this is to save them after some epochs.

tips: You will definitely **cry** if Colab crashes abruptly after 3 hours' training (trust me, it happens ~~a lot~~). All your progress will be gone if you do not back up. **SO SAVE YOUR MODEL FROM TIME TO TIME!!!**

```
torch.save({
  'model_state_dict': model.state_dict(),
  'optimizer_state_dict': optimizer.state_dict(),
  'scheduler_state_dict' : scheduler.state_dict(),
}, "gdrive/MyDrive/hw2p2/"+"Model_"+str(epoch))
```

An important concept to understand here is that what all are you saving in the model. You may only save the model if you want for inference but you need to save the entire state if you want to run it further. You might want to refer to the documentation for more understanding. https://pytorch.org/tutorials/beginner/saving_loading_models.html

Now, you can confidently retrieve your progress by re-loading your model and continue your adventure

```
temp = torch.load("gdrive/MyDrive/hw2p2/"+"Model_2")
network.load_state_dict(temp['model_state_dict'])
optimizer.load_state_dict(temp['optimizer_state_dict'])
scheduler.load_state_dict(temp['scheduler_state_dict'])
```

where `network` is the model name and you would have defined your optimizer and schedulers. Do not worry if you do not know what they are! You will get to know about these terms as the course starts.

Connecting to Kaggle in Colab

Install Kaggle API

For most of the assignments, the data would be huge for you to download and then upload it somewhere. The best option is to directly download the data from Kaggle every time you load the notebook.

This may sound troublesome but **this saves a lot of time.**

1. To begin with, you need to install Kaggle directly in the Colab VM

   ```
   !pip install kaggle
   !mkdir .kaggle
   ```

2. You would need to create a new API token in Kaggle as mentioned in the article below.

   https://towardsdatascience.com/setting-up-kaggle-in-google-colab-ebb281b61463

3. Once Kaggle is installed, you need to set up the Kaggle keys to let it identify your account of Kaggle with the current notebook settings.

```
import json
token = {"username":"your_username","key":"your_key"}
with open('/content/.kaggle/kaggle.json', 'w') as file:
    json.dump(token, file)
```

4. To set the correct token permissions, type the following.

```
!chmod 600 /content/.kaggle/kaggle.json
!cp /content/.kaggle/kaggle.json /root/.kaggle/
!kaggle config set -n path -v /content
```

Note: you may get errors after running the above commands, you can try running them again, or

```
import os, zipfile, tarfile, ipdb
os.environ['KAGGLE_USERNAME'] = "your_username"
os.environ['KAGGLE_KEY'] = "your_key"
```

Download data from Kaggle

Once you do the above steps, go to Kaggle competition and check the name of the competition from the URL. Run the command below to download the data from Kaggle.

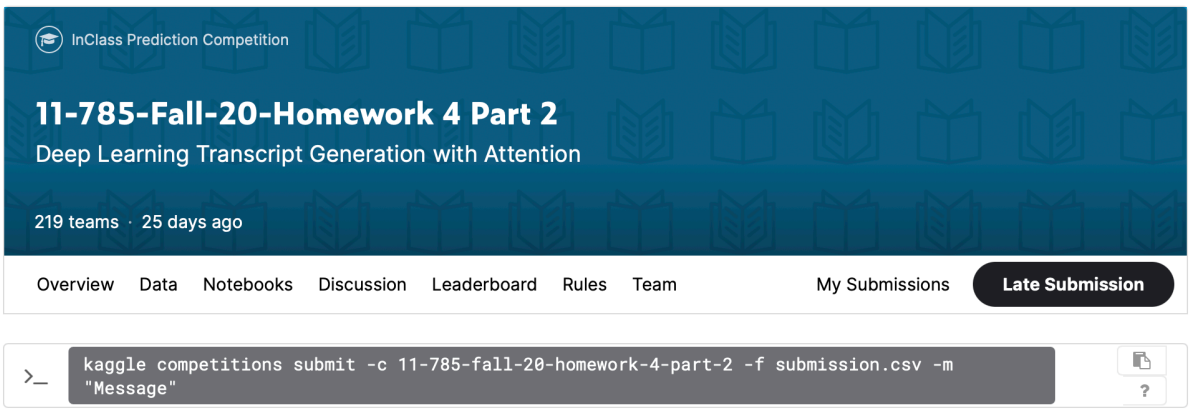
```
!kaggle competitions download -c 11-785-fall-20-homework-4-part-2
```

tips: even if you have unlimited storage for your Google Drive, you are **not** encouraged to download the dataset to your Drive due to speed and efficiency reasons. Otherwise, you may suffer from a long wait for the dataset to be unzipped.

Submit your results

You can either

1. download the `.csv` file and submit it manually
2. use the Kaggle API
    1. Go to the Kaggle competition page -> `Submission`



2. copy the command to your Colab Notebook and execute it

```
!kaggle competitions submit -c 11785-balabalabalabala
[YourFileName].csv -m "[Commit Message]"
```

3. check your score on the leaderboard

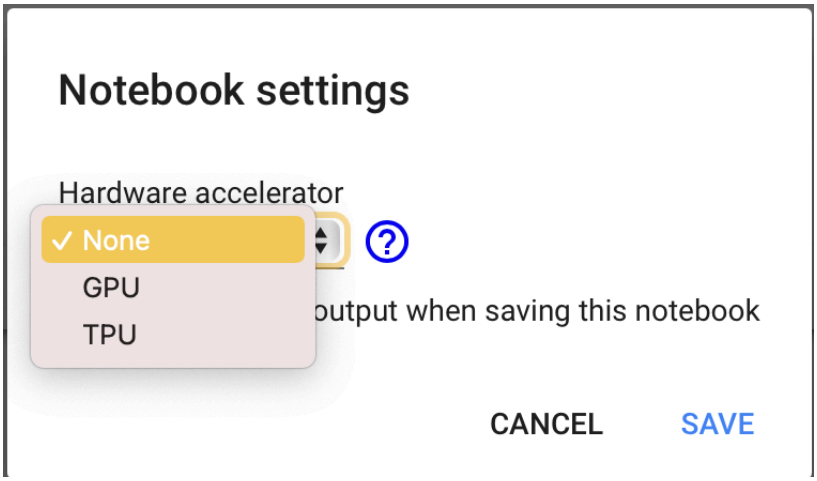
Limitations and Tips for Google Colab Users

"She was still too young to know that life never gives anything for nothing, and that a price is always exacted for what fate bestows. ---- Stefan Zweig, Marie Antoinette: The Portrait of an Average Woman"

Limitations [Please read carefully]

Colab is a savior, but it comes with a price.

1. You can only run **2** notebooks per google ID
2. There is a limitation per google ID in the GPU usage per 24 hours. So be wary about how much free GPU you are utilizing. If you have the notebook on but are not using it, change to a CPU or terminate the session if possible. (`Edit>Notebook settings`)



3. Once, you reach the usage limit, you will be banned for up to 24 hours from using any GPU.

   tips: you may want to use your other google IDs if you have
4. Also, if you want to run your models for a long time, **do not travel with your laptop**! Network disconnectivity will be frustrating with Colab. So, try to be in one place.
5. If you plan on leaving your model running for long hours unattended, Colab may disconnect. To prevent this, you can input the following command into the Chrome command console (`Ctrl+Shift+J` or google something similar for your browser/OS):

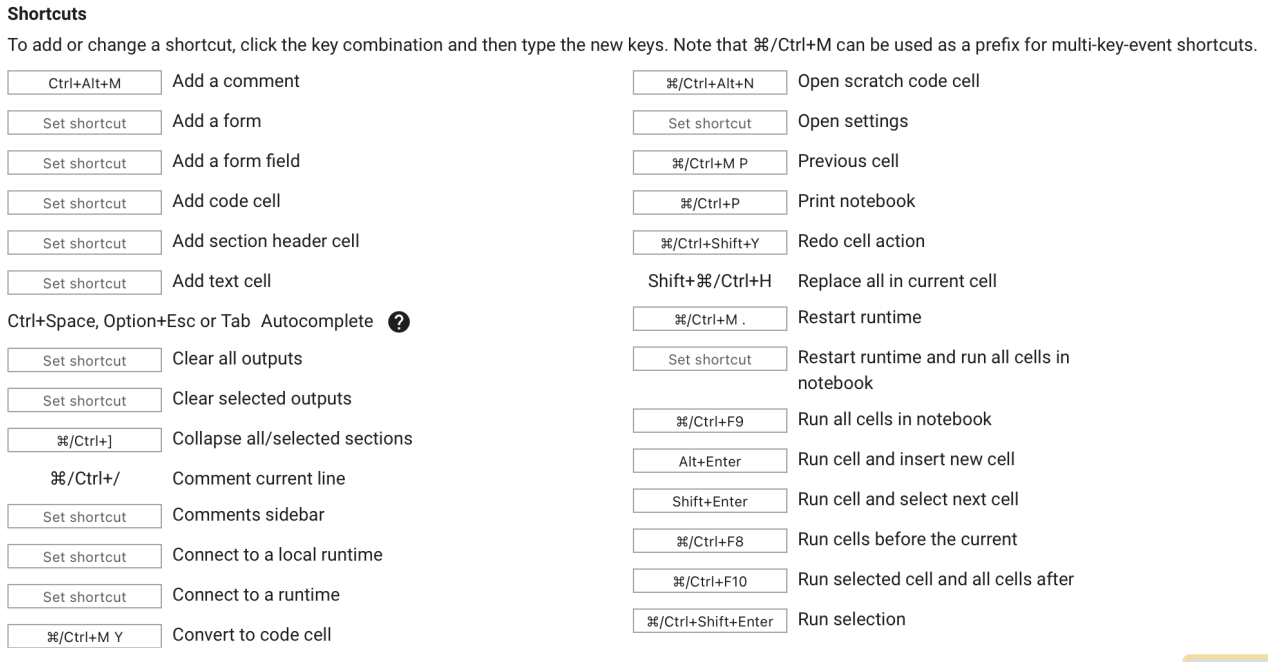
```
function ClickConnect(){
  console.log("Working");
  document.querySelector("colab-toolbar-button").click()
}setInterval(ClickConnect,600000)
```

6. ALL YOUR FILES WILL DISAPPEAR AS SOON AS YOU LEAVE Google Colab. So save them well.

Pro-tips

Keyboard shortcuts

You can use shortcuts to save time. `Tools > keyboard shortcuts` (`cmd/ctrl+M H`)



Upload/download files

You can upload files from your PC/Mac by running

```
from google.colab import files
files.upload()
```

and download a file by running

```
from google.colab import files
files.download('path/to/your/file')
```

Google Colab Pro

If AWS does not work for you and you cannot endure the limited GPUs that Colab delivers, you may turn to **Colab Pro**.

Disclaimer: All assignments can be finished without the Colab Pro subscription. That being said, if you start early and use resources wisely, you can definitely reach the A-cutoff. We, staff of 11-785, are only giving you another alternative. You are neither encouraged nor discouraged to use Colab Pro by any one of us. The Colab Pro rules are subject to change, and we, staff of 11-785, can do nothing about it.

Google Colab Pro provides priority access to faster GPUs, longer running notebooks and fewer idle timeouts, and more memory. Restrictions are subject to change by Google, not us. You can check the official website for more information.

https://colab.research.google.com/signup

tips: You **cannot** subscribe to Colab Pro with your Andrew email. You can use your own google account

- **Price: $9.99/month** Recurring billing. Remember to cancel it when you choose not to use it anymore.
- **GPUs: T4, P100** But you still have usage limits and available GPUs and TPUs may vary over time
- **Duration: 24 hrs** The notebook **may** stay connected for 24hrs, and idle timeouts are relatively lenient
- **Availability: the US and Canada ONLY** Use VPN if you are in another country/region

FAQ

- I run out of my Colab RAM. What should I do?
  - Colab only provides us a 12GB RAM, which may not be sufficient for your Deep Learning model training. What you can do is to simplify your model, or subscribe to Colab Pro
- I run out of my Colab Disk space. What should I do?
  - This can hardly happen if you write concise and efficient code. Please double check your code.
- I encountered CUDA: out of memory. What should I do?
  - Try decreasing your batch size
- I can not use Kaggle API to download the dataset
  - Check whether you join the competition and consent to the competition regulations
- I can not cd to a specific folder
  - Check where you are by `!pwd` and whether the folder exists by `!ls`. `%cd` can not generate a new folder for you and will return an error if it is asked to go to a folder that does not exist.