Notes on Using Google Colaboratory in AI Education

Mark J. Nelson
American University
Washington, DC, USA
mnelson@american.edu

Amy K. Hoover
New Jersey Institute of Technology
Newark, New Jersey, USA
ahoover@njit.edu

ABSTRACT

We discuss our experiences using Google Colaboratory (Colab), a hosted version of Jupyter Notebooks, in undergraduate artificial intelligence (AI) courses at two universities. Colab was designed for AI and data science researchers to share reproducible experiments and explanations of techniques, but we have also found it well suited to classroom use. The primary benefit is that it provides students computational resources sufficient to run modern AI techniques interactively, and avoids students needing to separately configure software packages and dependencies, since they can run notebooks shared by the instructor. We briefly outline two of our notebooks, for teaching deep learning with Tensorflow, and reinforcement learning with OpenAI Gym.

CCS CONCEPTS

• **Social and professional topics** → **Computing education**.

KEYWORDS

Google Colab, Jupyter Notebook, notebook interface, AI education

ACM Reference Format:
Mark J. Nelson and Amy K. Hoover. 2020. Notes on Using Google Colaboratory in AI Education. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '20), June 15–19, 2020, Trondheim, Norway.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3341525.3393997

1 INTRODUCTION

A hands-on approach is at the core of experiential learning, but a challenge in designing courses in modern artificial intelligence (AI) is choosing appropriate tools and frameworks for in-class examples and homework assignments. It is not only important that the tools are able to cover the depth of the material presented in the course, but simple enough for instructors to implement in the classroom and students to maintain focus on the curriculum. Jupyter Notebooks (i.e., interactive documents interleaved with text and code) are an increasingly popular tool for researchers to conduct and communicate research findings, yet simple enough for developing classroom examples and homework [1, 2]. However these notebooks still require sophisticated hardware to run many of the approaches popular in AI today.

ITiCSE '20, June 15–19, 2020, Trondheim, Norway

ACM ISBN 978-1-4503-6874-2/20/06.
https://doi.org/10.1145/3341525.3393997

Offered as a free service from Google, Colaboratory (or Colab)[1] provides a Jupyter Notebook interface with access to Google hardware. Notebooks are run in Linux-based virtual machines (VMs) provided and maintained by Google where computation can be performed with central processing units (CPUs), or accelerated through specialized graphical processing units (GPUs) and tensor processing units (TPUs). Available hardware for each VM varies by session, but typically includes top-of-the-line of NVIDIA GPUs (K80, T4, or P100), around 8–12GB of RAM, and 50–70 GB of free space on the VM hard drive. Since Colab notebooks are intended for interactive use rather than long-running experiments, VMs disconnect after an idle timeout, and limit sessions to 12 hours.

Besides providing computational resources, the cloud-based VMs backing Colab notebooks are pre-loaded with common AI packages (numpy, torch, tensorflow, etc.). While a potentially attractive alternative to Colab is a framework for managing packages (e.g., Anaconda), they often overwhelm those students already under heavy cognitive load. Running in-class examples that rely on a variety of Python package dependencies presents configuration and debugging challenges for each student's operating system and general environment (e.g., "dependency hell"). In our experience, students can begin to perceive simply installing and configuring packages to be the main content of the course rather than a limitation of the tools for experiential learning. While even the Colab VMs occasionally need additional updating, the standardized environment significantly reduces the burden on the instructor and lets us focus classroom time on the course material.

As Colab notebooks are hosted Jupyter Notebooks, they follow a typical notebook workflow that intermixes narrative explanations and code-based interactive demonstrations [3]. Besides gaining some use in AI education [2], notebook-based workflows are becoming an popular medium for AI and data science researchers to disseminate their new approaches and results. Our hope therefore is that not only is Colab convenient for us as instructors, but also helps prepare students for research and academic environments that often require more advanced navigation of these workflows.

While Jupyter Notebooks, package managers, containers, etc., are already present in some classrooms, we've found the combination of easy-to-use tools provided by Google Colab to provide an ideal environment for classroom instruction. This paper introduces Google Colab from our perspective as instructors at two different American universities, teaching undergraduate AI courses of around 30 students each. Over the past year we have been using Colab for in-class demos and homework assignments in these classes. We discuss a few examples of the notebooks we've prepared, and end with a discussion of some challenges and potential risks in relying on Colab as an educational platform.

[1] https://colab.research.google.com

2 EXAMPLE NOTEBOOKS

We have developed notebooks for in-class demos and homework assignments that use a range of AI techniques: evolutionary computation, planning, machine learning, etc. Two examples are:

1. Deep Learning with Tensorflow. Deep neural networks are a prominent machine learning approach that have recently begun to be integrated into undergraduate AI curricula [4]. We introduce students to neural network architecture, training, and inference, using the Tensorflow framework [5] with GPU acceleration. Both Tensorflow 1.x and 2.x are pre-configured (version chosen on a per-notebook basis), along with GPU drivers.

We walk students through how to train networks, report performance, and analyze results. We start with the standard MNIST dataset of handwritten digits, since it can be trained at interactive speeds live in class, and both the data and neuron activations can be displayed visually as images inside the notebook. We use matplotlib to plot training curves within the notebook. The neural-network segment is further eased by the official Tensorflow documentation being available as Colab notebooks.[2]

2. Reinforcement Learning with OpenAI Gym. OpenAI Gym [6] wraps a set of reinforcement learning (RL) problems, i.e. environments in which an agent needs to learn how to act based on experience. These include classic control problems, game-playing in Atari games (which wraps an Atari emulator), learning to walk with bipedal robots, etc. It comes mostly preconfigured on Colab, with the exception of environments that require the commercial MuJoCo physics engine.

We start with Atari games, using the variants where observations are pixels. Since observations in these environments are a pixel array of the game screen, we can display states inline in the notebook. Manually stepping through taking actions in different game states and displaying the results gives intuition for how RL environments work: states, observations, actions, successor states, and so on (Figure 1). For environments where the observations aren't pixel grids, it does require more adaptation to show visuals within a notebook: OpenAI Gym expects to be able to write to a graphical display, but this can be hacked to output to a video saved on the Colab VM instead, which can then be played back in the notebook.

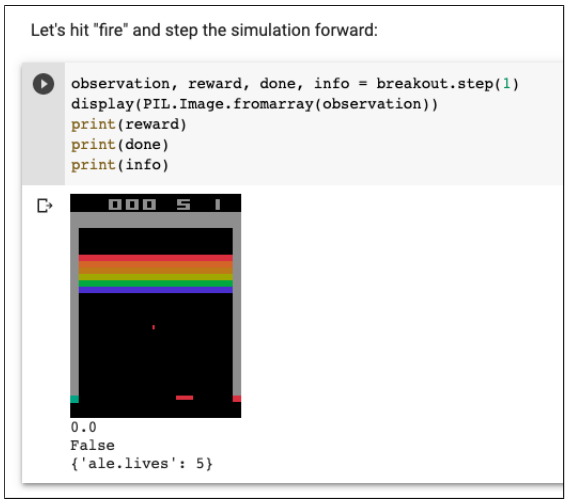

Figure 1: Example of interacting with an OpenAI Gym environment in a notebook. Here, we single-step through taking actions in the Atari game *Breakout*.

3 DISCUSSION AND CONCLUSION

Google Colaboratory (i.e., Colab) is a free service that combines the features of Jupyter Notebooks with Google-hosted VMs and high-end hardware. Ideal for classrooms designed with experiental learning in mind, instructors can easily share read-only in-class examples, activities, and homework as notebook files that students can execute without additional package management or workflow setup. As AI research is increasingly disseminated through the notebook format in part for reproducibility, experience with this workflow may also prepare students for future careers in either industry or academia.

The primary risk of using a free hosted platform like Colab is that it might be discontinued or turn into a paid service. This is partly mitigated by the fact that Colab is built on Jupyter Notebooks, meaning that there is minimal software lock-in. One possible migration path is to hosted Jupyter notebooks from other providers, such as Microsoft Azure Notebooks or IBM Watson Studio.[3]

With additional instructor burden, it is also possible to host Jupyter notebooks in VMs on local university equipment, using the open-source JupyterHub platform.[4] The most difficult issue in migration would be reworking notebooks and assignments that rely on every student having access to a recent GPU – it's unlikely that these migration routes would be able to provide that level of hardware resources with the funding we have available.

REFERENCES

[1] Thomas Kluyver et al. 2016. Jupyter notebooks: a publishing format for reproducible computational workflows. In *Proceedings of the International Conference on Electronic Publishing.*

[2] Keith J. O'Hara, Douglas Blank, and James Marshall. 2015. Computational notebooks for AI education. In *Proceedings of the 2015 International Flairs Conference.*

[3] Alistair Willis, Patricia Charlton, and Tony Hirst. 2020. Developing students' written communication skills with Jupyter notebooks. In *Proceedings of the SIGCSE Technical Symposium on Computer Science Education.*

[4] Amy K. Hoover, Adam Spryszynski, and Michael Halper. 2019. Deep learning in the IT curriculum. In *Proceedings of the SIGITE Conference on Information Technology Education.*

[5] Martín Abadi et al. 2016. Tensorflow: a system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation.*

[6] Greg Brockman et al. 2016. OpenAI Gym. arXiv: 1606.01540.

[2] https://www.tensorflow.org/tutorials

[3] We didn't start with these providers due to more complicated signup, less generous free tiers, and the fact that Colab is in fairly widespread use among AI researchers. They otherwise share some of the same advantages.

[4] https://github.com/jupyterhub/jupyterhub