CS231n Convolutional Neural Networks for Visual Recognition

Course Website

Assignment 3

This assignment is due on **Wednesday, May 27 2020** at 11:59pm PDT.

▶ Handy Download Links

Goals

In this assignment, you will implement recurrent neural networks and apply them to image captioning on the Microsoft COCO data. You will also explore methods for visualizing the features of a pretrained model on ImageNet, and use this model to implement Style Transfer. Finally, you will train a Generative Adversarial Network to generate images that look like a training dataset!

The goals of this assignment are as follows:

- Understand the architecture of recurrent neural networks (RNNs) and how they operate on sequences by sharing weights over time.
- Understand and implement both Vanilla RNNs and Long-Short Term Memory (LSTM) networks.
- Understand how to combine convolutional neural nets and recurrent nets to implement an image captioning system.

- Explore various applications of image gradients, including saliency maps, fooling images, class visualizations.
- Understand and implement techniques for image style transfer.
- Understand how to train and implement a Generative Adversarial Network (GAN) to produce images that resemble samples from a dataset.

Setup

You should be able to use your setup from assignments 1 and 2.

You can work on the assignment in one of two ways: **remotely** on Google Colaboratory or **locally** on your own machine.

Regardless of the method chosen, ensure you have followed the setup instructions before proceeding.

Option A: Google Colaboratory (Recommended)

Download. Starter code containing Colab notebooks can be downloaded here.

If you choose to work with Google Colab, please familiarize yourself with the recommended workflow.

CS231n Google Colab Assignment Workflow Tutorial
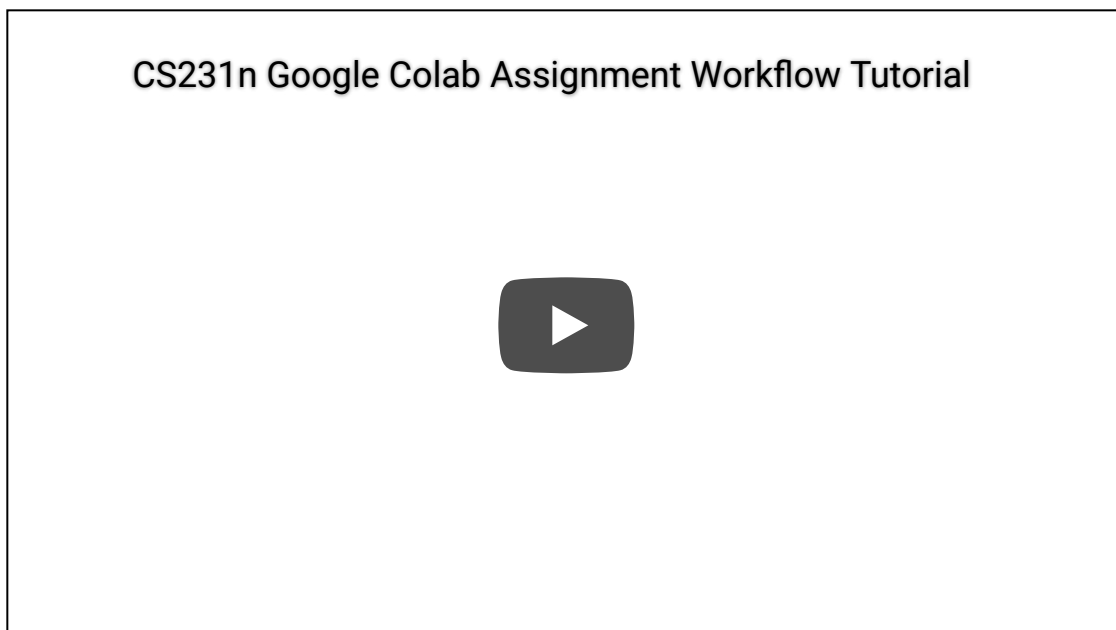
Note. Ensure you are periodically saving your notebook (`File -> Save`) so that you don't lose your progress if you step away from the assignment and the Colab VM disconnects.

Once you have completed all Colab notebooks **except** `collect_submission.ipynb`, proceed to the submission instructions.

Option B: Local Development

Download. Starter code containing jupyter notebooks can be downloaded here.

Install Packages. Once you have the starter code, activate your environment (the one you installed in the Software Setup page) and run `pip install -r requirements.txt`.

Download data. Next, you will need to download the COCO captioning data, a pretrained SqueezeNet model (for TensorFlow), and a few ImageNet validation images. Run the following from the `assignment3` directory:

```
cd cs231n/datasets
./get_datasets.sh
```

Start Jupyter Server. After you've downloaded the data, you can start the Jupyter server from the `assignment3` directory by executing `jupyter notebook` in your terminal.

Complete each notebook, then once you are done, go to the submission instructions.

You can do Questions 3, 4, and 5 in TensorFlow or PyTorch. There are two versions of each of these notebooks, one for TensorFlow and one for PyTorch. No extra credit will be awarded if you do a question in both TensorFlow and PyTorch

Q1: Image Captioning with Vanilla RNNs (29 points)

The notebook `RNN_Captioning.ipynb` will walk you through the implementation of an image captioning system on MS-COCO using vanilla recurrent networks.

Q2: Image Captioning with LSTMs (23 points)

The notebook `LSTM_Captioning.ipynb` will walk you through the implementation of Long-Short Term Memory (LSTM) RNNs, and apply them to image captioning on MS-COCO.

Q3: Network Visualization: Saliency maps, Class Visualization, and Fooling Images (15 points)

The notebooks `NetworkVisualization-TensorFlow.ipynb`, and `NetworkVisualization-PyTorch.ipynb` will introduce the pretrained SqueezeNet model, compute gradients with respect to images, and use them to produce saliency maps and fooling images. Please complete only one of the notebooks (TensorFlow or PyTorch). No extra credit will be awardeded if you complete both notebooks.

Q4: Style Transfer (15 points)

In thenotebooks `StyleTransfer-TensorFlow.ipynb` or `StyleTransfer-PyTorch.ipynb` you will learn how to create images with the content of one image but the style of another. Please complete only one of the notebooks (TensorFlow or PyTorch). No extra credit will be awardeded if you complete both notebooks.

Q5: Generative Adversarial Networks (15 points)

In the notebooks `GANS-TensorFlow.ipynb` or `GANS-PyTorch.ipynb` you will learn how to generate images that match a training dataset, and use these models to improve classifier performance when training on a large amount of unlabeled data and a small amount of labeled data. Please complete only one of the notebooks (TensorFlow or PyTorch). No extra credit will be awarded if you complete both notebooks.

Submitting your work

Important. Please make sure that the submitted notebooks have been run and the cell outputs are visible.

Once you have completed all notebooks and filled out the necessary code, there are **two** steps you must follow to submit your assignment:

1. If you selected Option A and worked on the assignment in Colab, open `collect_submission.ipynb` in Colab and execute the notebook cells. If you selected Option B and worked on the assignment locally, run the bash script in `assignment3` by executing `bash collectSubmission.sh`.

This notebook/script will:

- Generate a zip file of your code (`.py` and `.ipynb`) called `a3.zip`.
- Convert all notebooks into a single PDF file.

Note for Option B users. You must have (a) `nbconvert` installed with Pandoc and Tex support and (b) `PyPDF2` installed to successfully convert your notebooks to a PDF file. Please follow these installation instructions to install (a) and run `pip install PyPDF2` to install (b). If you are, for some inexplicable reason, unable to successfully install the above dependencies, you can manually convert each jupyter notebook to HTML (`File -> Download as -> HTML (.html)`), save the HTML page as a PDF, then concatenate all the PDFs into a single PDF submission using your favorite PDF viewer.

If your submission for this step was successful, you should see the following display message:

`### Done! Please submit a3.zip and the pdfs to Gradescope. ###`

2. Submit the PDF and the zip file to Gradescope.

Note for Option A users. Remember to download `a3.zip` and `assignment.pdf` locally before submitting to Gradescope.

---

 cs231n
 cs231n