Docstring to Code

Ashkon Aghassi, Cindy Lay, Frederick Qin

April 2021

Introductions

Ashkon Aghassi

Student

•••

Education & Experience

- Junior at Claremont McKenna College (Class of 2022) pursuing B.A. in Economics and Computer Science
- Lover of skiing and surfing

Frederick Qin

Student

•••

Education & Experience

- Junior at Claremont McKenna College (Class of 2022) pursuing B.A. in Economics and Computer Science

Cindy Lay

Student

•••

Education & Experience

- Junior at Claremont McKenna College (Class of 2022) pursuing B.A in Computer Science and Mathematics

Agenda

I. Palate Cleanser

II. Motivation

III. Libraries, data, resources

IV. Progress

V. Demo

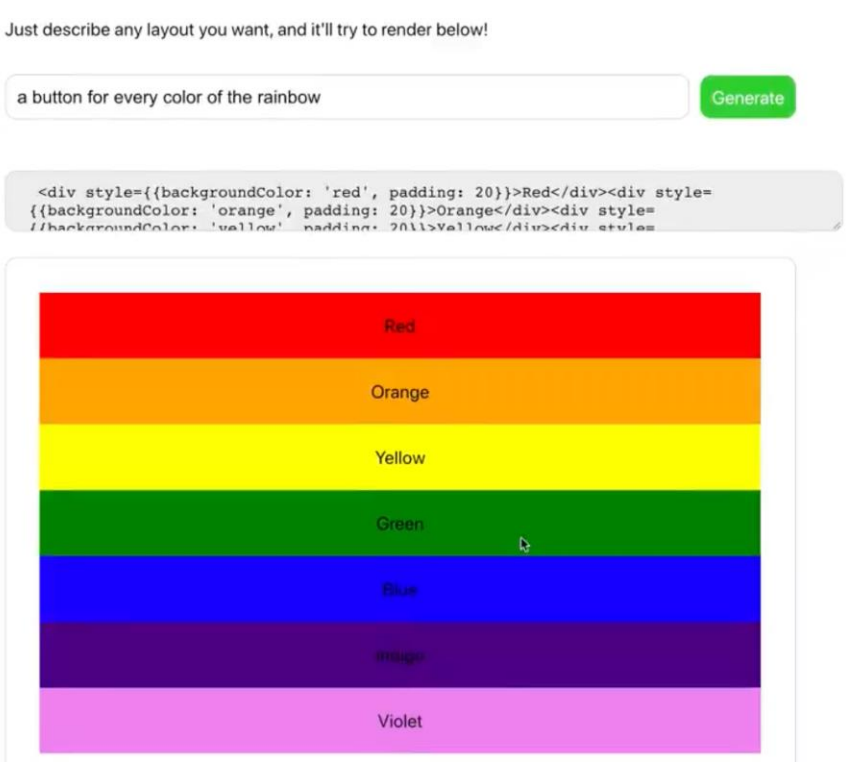
VI. Future Plans

Speaking of Python…

Project Motivation

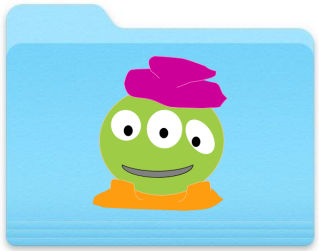
Our project explores the ability of the GPT-2 pre-trained Neural Network to write code given a well-written docstring

- CS5 taught us that docstrings are important!!
  - We wanted to put our well-written docstrings to the test
- GPT-3
  - We liked GPT-3 but didn't have a formal research proposal
  - It is so powerful you can write fake news
  - https://twitter.com/sharifshameem/status/1282676454690451457?s=20
- Interested in NLP and Neural Networks:
  - We used neural networks to build models converting math handwriting-to-latex and detect similar artwork
  - Our NLP homeworks sparked interest in open source
- Are we coding away our own jobs?
  - Less of a need for programming if we can generate code



Libraries, Data, Resources

Depicted is our complete ML pipeline to create our own python-writing language-generation model.



CS5 Files

Python
OS Module

Preprocess

Intermediate Data Representation

GPT-2
SMALL

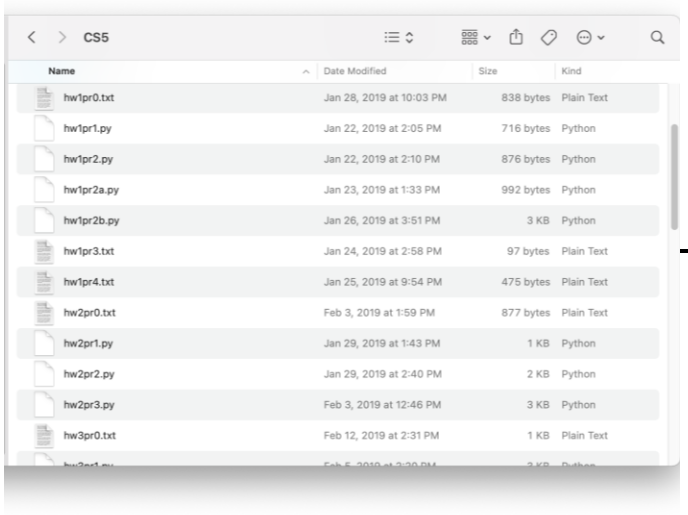
117M Parameters

Finetune Model

Current Progress, Preprocessing

Our preprocessing step takes any directory path, locates all the python files within that directory, identifies python functions (and their docstrings). It then writes it to our desired specific format.

Data Preprocessing




```
 <|endoftext|>
"""Return value: tpl returns thrice its argument
       Argument x: a number (int or float)
    """
def tpl(x):
    """Return value: tpl returns thrice its argument
       Argument x: a number (int or float)
    """
    return 3*x


 <|endoftext|>
"""Returns the square of its argument
       Argument x: a number (int of float)
    """
def sq(x):
    """Returns the square of its argument
       Argument x: a number (int of float)
    """
    return x*x
```

Key Takeaways:

- Our filesystem traversing scripts from the beginning of class were super handy!
- Regex is both extremely powerful, yet really annoying :)
- Python makes reading and writing from files very convenient
- We were able to identify 292 different functions and their docstrings

Current Progress, Data & Modeling

To train our own text generation model, all we need is to feed in formatted text into a super fancy (big) gpt-2 model.

Data & Modeling

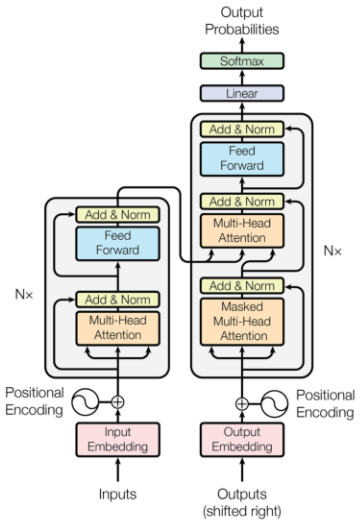
Specific ending character for gpt-2

Docstring: our desired seed

Function: our desired output from the model

```
<|endoftext|>
"""Return value: tpl returns thrice its argument
    Argument x: a number (int or float)
    """
def tpl(x):
    """Return value: tpl returns thrice its argument
        Argument x: a number (int or float)
    """
    return 3*x

<|endoftext|>
"""Returns the square of its argument
    Argument x: a number (int of float)
    """
def sq(x):
    """Returns the square of its argument
        Argument x: a number (int of float)
    """
    return x*x
```



Process:

Token 1 → *Predict* Token 2

.
.
.

Token *n* → *Predict*

Key Takeaways:

- GPT2 (and GPT3) are massive and NOT easy to make but they ARE easy to finetune!
- Google Colab is an absolute nightmare (but we love and NEED free GPUs)
- Our 292 functions created 79124 tokens when encoded, and we trained for 1024 epochs (so far)
- Our jobs as software writers are secure for the near future (you will see)

Demo (Screenshots)

To say the least, the model behaves …. interestingly. Here are 2 examples with temperature = 1 and top-k = 40.

```
"""Result: quad returns four times its argument
       Argument x: a number (int or float)
       Spam is great, and quad("spam") is better!
    """
2021-04-25 05:06:06.586473: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10.0
====================================== SAMPLE 1 ======================================

def quad(x,y):
    """Result: quad returns four times its argument
       Argument x: a number (int or float)
       Spam is great, and quad("spam") is better!
    """
    return x*x + y*y
```

```
"""Returns a list of 10 random numbers"""
2021-04-25 05:24:11.937031: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10.0
====================================== SAMPLE 1 ======================================

def randomRows(numRows):
    """Returns a list of 10 random numbers"""
    d = len(Rows)#10
    numbers = rand()
    s = int(s[0])#4
    sdec = int(s[1])#2
    sgold = '' # added to save space
    # loop over the 100-character-wide list r_start and r_end, returning:
       numRows,
       s, and
       sdec,
       s_start, and s_start
    for r in range(100):
        s += ' \n '
       # look for double-spaces in the start of the row
    # we just created
    return numbers/2
```

Live Demo

Future Work

As of now, the model is not yet close to being able to understand docstrings and writing code :(
We have not ruled out the possibility, yet!

Future Work for This Class

- Optimize modeling hyperparameters
  - Try training for more epochs
  - Experiment with Adam optimizer
- Play around with generation hyperparameters
  - Top-k
  - Temperature
- Better understand data
  - Look at consistency of docstring formatting
  - … as well as code
  - See if providing more uniform data gives better results
- Frontend UI?

Future Work Down the Road

- GPT-3 … bigger and better!

screenshot this for hw12_835talk

What are your names again?

Ashkon, Cindy, & Fred

What stands out as something you learned from / liked about this presentation?

The icons at the corners, especially at the bottom right.

If you were presenting this, what's something you'd consider adding/updating?

Nothing. It was perfect.

Share a thought or suggestion you have on the project itself – not the presentation – a point of interest, question, suggestion, etc.

It's missing the number 42 :(

Sources:
https://github.com/nshepperd/gpt-2
https://medium.com/ai-innovation/beginners-guide-to-retrain-gpt-2-117m-to-generate-custom-text-content-8bb5363d8b7f
https://github.com/openai/gpt-2