

Assignment 4 - Creating a StFX focused Chat-GPT

This assignment shows you how easy it is to use OpenAI's GPT-3 API on your own documents. The GPT-3 language model was trained how to speak in 2021 using content in the world at that time. It has been two years and a few things have changed at StFX. We want to create our own GPT-3 chatbot, but teach it about the new material at StFX. Specifically, we want it to know about Schwartz programming.

The hard part of this assignment is to learn how to use git for making changes to code. The fun part is creating a Chatbot!.

To begin this assignment, you will need to have the following:

- **Python 3** and pip installed on your laptop. With Python 3 and Pip installed
- **Git** software must be installed (installed already on most mac/linux machines) [Learn more here](#).
- A free **GitHub** account (github.com)

Step 1: Create a python virtual environment (Optional but encouraged)

You can learn more about python virtual environments [here] (<https://docs.python.org/3/library/venv.html>), but basically, you are going to install some 3rd part libraries and typically wouldn't want to conflict with other projects you are working on, so you create a virtual environment (like a sandbox) where these packages will be installed:

Let's create a virtual environment in your home directory:

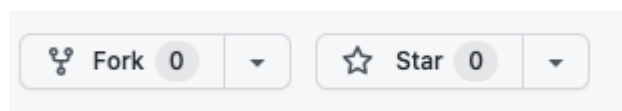
```
mkdir -p ~/environments
python -m venv ~/environments/bsad487Assignment4
source ~/environments/bsad487Assignment4/bin/activate
```

Step 2: Get a developer personal access token for Git.

In order to login to your Git account from the command line terminal, you will need a personal access token generated on GitHub - this is basically a time-limited password. You can find a [tutorial here] (<https://www.theserverside.com/blog/Coffee-Talk-Java-News-Stories-and-Opinions/How-to-create-a-GitHub-Personal-Access-Token-example>). Copy that password somewhere safe and easy to access - you will need to use it a few times.

Step 3: Get the starter code

Open a browser, login to github.com, and navigate to the github repo for [this assignment] (https://github.com/dmattie/bsad487_A4).



Click "Fork" in the top right.

You'll now have your own copy of that repository in your github account.

Open a command line terminal and change the directory to your Documents directory. For example, on OSX terminal, type the following: `cd ~/Documents`

On Windows, type the following: `cd %userprofile%\Documents`

Perform a git pull clone the Git Repository I created for this assignment:

```
git clone git@github.com:**username**/bsad487_A4
```

where **username** is your username

Step 4: Install dependencies and Set the API Key

To run the code you just downloaded, you will need to install a few packages required by the code. Run the following command line terminal to install dependencies:

```
pip install -r requirements.txt
```

If you receive an error asking to update pip, please do as instructed and repeat the command above.

We are almost ready to run the code, but first you have to enter my OpenAI API key provided on Moodle. This is a key tied to my credit card, and every time you run the code, I get charged a little bit - so I won't leave the key inside the code, otherwise the world gets it and uses free GPT calls.

Using a text editor like vim, nano, Notepad++, or vscode -- Replace the key on line 9 of the file **my-x-gpt.py** with the new key:

```
os.environ['OPENAI_API_KEY'] = 'sk-#####'
```

Eg.

```
os.environ['OPENAI_API_KEY'] = 'sk-ABCDEF.....get...from...moodle'
```

Step 5: Let's run the code

Type the following to run the program:

```
python my-x-gpt.py
```

This uses the python interpreter to run the code found in my-x-gpt.py

The code performs the following actions: Read through the files stored in the documents directory (there are three originally, you may contribute more). I have a file on the new NSCC-StFX degree program in Business Intelligence in Analytics, one on the Marketing program, and one on what its like to study at StFX.

Create a vectorized index of words found in those documents. This will provide an in-memory model to help GPT-3 answer the question with the new information.

Offer you a change to ask a question

Use OpenAI's GPT-3 language model to answer the question using existing knowledge or from the new information found in the documents we just added.

Example: Ask the following:

```
Tell me about the new NSCC-StFX Business Intelligence Degree
```

```
Tell me about Marketing!
```

Step 6: Add a new document

Now it's time to teach the model something new. Copy-paste content from any page on business.stfx.ca or stfx.ca and put into a basic text file (.txt) in the documents subdirectory of the code.

Re-run your program, and ask my-x-gpt a question that can be answered with your new content. See what happens!

Step 7: Commit your code and push it up

There are three commands to save your work and push back to GitHub:

```
git add .
```

This means: Add any changes to a set (index) of files to be pushed up. "." means all changes.

```
git commit -m "my recent changes message"
git push
```

Email me a link (github URL) to your repo if you get this far to signify completing assignment 4.