

Project: Use ChatGPT to build an Ubuntu system that can answer questions about your website

Md Shahadat Hossain Khan (19609)

TA of Machine Learning (CS550)

Here we are going to implement an automated chatting system exclusively for our website using OpenAI. Our new system will answer questions with natural language as ChatGPT use. The new system will answer from our website content.

To achieve this, we need to train OpenAI with our website data. So, are going to crawl our website first to get the data, then we will pass those data OpenAI API to train. This API will return some formatted data that we will save as processed data. When user ask any question, we will pass this processed data to OpenAI API and that API process that data to prepare answer. So we have three part of our program, they are -

Crawl: to get data from website for train

Prepare: to prepare those data and save the prepared data as processed data to use in future

Ask: to get answer of user's question

I merge 1 & 2 into one file and quiring OpenAI into 3rd file. We will use docker to package our python program. So, we can easily maintain our program.

Also we need to prepare our static website from which OpenAI will find the answer. For this purpose I create a static site and host it with this project.

For this project we have another challenge, we use HTTPS while crawling. So, I need to host it with SSL. Here I use Let's encrypt free SSL to achieve this. As we using micro-server, we need "standalone" SSL.

```
$ sudo certbot certonly --standalone -d  
python.site4chatgptrnd.shahadathossain.com
```

Note: Before execute above code we need to ensure that 80 and 443 port is not bind with the domain.

When it run successfully, it provide all necessary certificate files. In my server the location of those files was "/etc/letsencrypt/live/python.site4chatgptrnd.shahadathossain.com/" Now, we need to copy "cert.pem", "" and "" into our "<project-root>/ssl-certificates/" directory. To do this, we can apply following commands -

```
$ sudo cp  
/etc/letsencrypt/live/python.site4chatgptrnd.shahadathossain.com/fullchain.pem  
pure-python-version/ssl-certificates/  
$ sudo cp  
/etc/letsencrypt/live/python.site4chatgptrnd.shahadathossain.com/cert.pem  
pure-python-version/ssl-certificates/  
$ sudo cp  
/etc/letsencrypt/live/python.site4chatgptrnd.shahadathossain.com/privkey.pem  
pure-python-version/ssl-certificates/
```

Note: Symbolic-link or hard-link will not work here in Docker.

As we encapsulate our code into Docker, our challenge is how we point the certificates with our Docker file system. We can create volume and point that folder. But, I can't did it successfully. So, I do another way, which is copy the certificates each time docker image build. So, problem is, I need to rebuild our image each time SSL certificates need to renew. Also after renew SSL certificate, we need to execute above copy commands.

Another thing we need to do is to setup an OpenAI API Key by visiting <https://platform.openai.com/docs/api-reference/introduction> or <https://platform.openai.com/account/api-keys> [through your signed in account > API Keys link] Please note, you need to spend a small amount of money like \$5.00 to get access of OpenAI for three months.

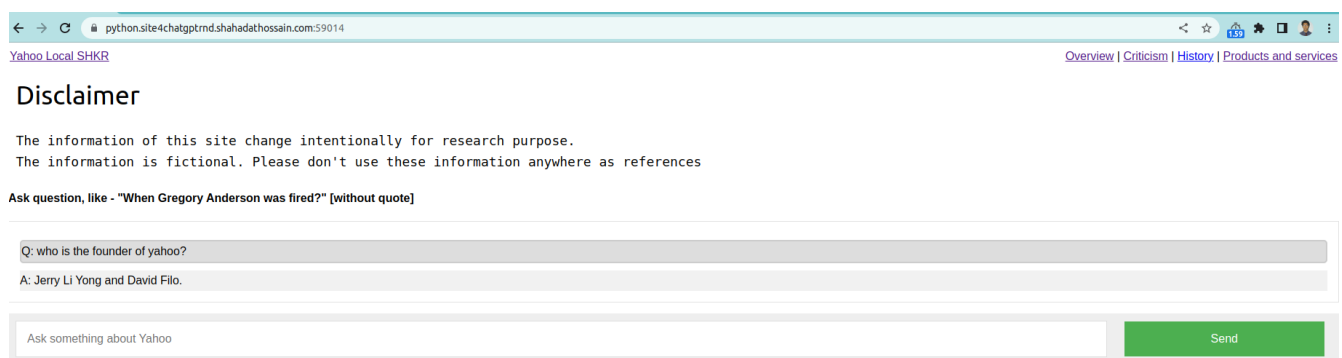
We can interact with the API through HTTP requests from any language, via OpenAI's official Python bindings, OpenAI's official Node.js library, or a community-maintained library. I'm Python fan, so I'll use Python bindings. As I packaged into docker, so "Docker" file take care installation / setup process. To run build docker and run docker you can use these code -

```
$ sudo docker build -t openai2test .  
$ sudo docker run -d --name openaiusingpurepython_container --network=host  
openaiusingpurepython:latest
```

After running docker, we need to crawl our project and execute learning session of AI. We need to invoke following commands into our running container -

```
$ sudo docker exec -it openaiusingpurepython_container python ./crawl-and-  
train.py
```

Now we are ready to run prescribed code of OpenAI. We already build and execute our docker and it will communicate through 59014 port at localhost.



Please note that I change the spelling of "Jerry Li Yang" and "David Filo" as a proof that OpenAI read from my fictional website.

GitHub Link: <https://github.com/mkhan-sfbu/openai4website>