

This is week's 5 Data Glacier internship deliverable.

I will be using week'4 model and data.

This work was submitted on 18/07/2021 to Data Glacier by me Nada Belaidi.

I will be using fastapi from uvicorn so I started with installing them to the virtual environment I used for last week's flask app (<https://github.com/NadaBelaidi/Strawberry-segmentation-simple-flask-app>)

```
(venv) C:\Users\Ahmed\Desktop\Strawberry-segmentation - Copie>py -m pip install uvicorn
Collecting uvicorn
  Using cached uvicorn-0.14.0-py3-none-any.whl (50 kB)
Requirement already satisfied: click>=7.* in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (from uvicorn) (8.0.1)
Collecting asgiref>=3.3.4
  Using cached asgiref-3.4.1-py3-none-any.whl (25 kB)
Collecting h11>=0.8
  Using cached h11-0.12.0-py3-none-any.whl (54 kB)
Requirement already satisfied: colorama in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (from click>=7.*->uvicorn) (0.4.4)
Installing collected packages: h11, asgiref, uvicorn
  WARNING: The script uvicorn.exe is installed in 'C:\Users\Ahmed\AppData\Local\Programs\Python\Python39\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed asgiref-3.4.1 h11-0.12.0 uvicorn-0.14.0

(venv) C:\Users\Ahmed\Desktop\Strawberry-segmentation - Copie>py -m pip install fastapi
Requirement already satisfied: fastapi in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (0.66.0)
Requirement already satisfied: starlette==0.14.2 in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (from fastapi) (0.14.2)
Requirement already satisfied: pydantic!=1.7,!<1.7.1,!<1.7.2,!<1.7.3,!<1.8,!<1.8.1,<2.0.0,>=1.6.2 in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (from fastapi) (1.8.2)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\ahmed\appdata\local\programs\python\python39\lib\site-packages (from pydantic!=1.7,!<1.7.1,!<1.7.2,!<1.7.3,!<1.8,!<1.8.1,<2.0.0,>=1.6.2->fastapi) (3.7.4.3)
```

Since my deep learning model requires images to run on, I needed python-multipart in order to upload image files.

```
(venv) PS C:\Users\Ahmed\Desktop\Strawberry-segmentation> py -m pip install python-multipart
WARNING: Ignoring invalid distribution -ensorflow (c:\users\ahmed\desktop\strawberry-segmentation\venv\lib\site-packages)
WARNING: Ignoring invalid distribution -ensorflow (c:\users\ahmed\desktop\strawberry-segmentation\venv\lib\site-packages)
Collecting python-multipart
  Downloading python-multipart-0.0.5.tar.gz (32 kB)
Requirement already satisfied: six>=1.4.0 in c:\users\ahmed\desktop\strawberry-segmentation\venv\lib\site-packages (from python-multipart) (1.15.0)
Building wheels for collected packages: python-multipart
  Building wheel for python-multipart (setup.py) ... done
  Created wheel for python-multipart: filename=python_multipart-0.0.5-py3-none-any.whl size=31670 sha256=c60a57a2b6c917cdb00192501c10d8c6dd3f7f8b2023e5ffff3d7f61fbf20589
  Stored in directory: c:\users\ahmed\appdata\local\pip\cache\wheels\fe\04\d1\1a10661cc45f03c3cecd50deb2d2c22f57b4e84a75b2a5987e
```

I will be then importing modules and libraries needed.

```
from __future__ import division, print_function
# coding=utf-8
import sys
import os
import glob
import re
import numpy as np
import uvicorn
from fastapi import FastAPI, Form, File, UploadFile
from tensorflow import keras
import h5py
# Keras
from keras.applications.imagenet_utils import preprocess_input, decode_predictions
from keras.models import load_model
from keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
from tensorflow.keras.preprocessing import image as image_utils
from tensorflow.keras.applications.imagenet_utils import preprocess_input
```

Creating the app and loading the model.

```
app = FastAPI()
# Model saved with Keras model.save()
MODEL_PATH = (r'C:\Users\Ahmed\Desktop\Strawberry-segmentation - Copie\models')

# Load your trained model
model = load_model(MODEL_PATH)
model.make_predict_function()
print('Model loaded. Check http://127.0.0.1:5000/)
```

Creating the model_predict function.

```
def model_predict(image_path):
    image = image_utils.load_img(image_path, target_size=(224, 224))
    image = keras.preprocessing.image.img_to_array(image)
    image = image.reshape(1,224,224,3)
    image = preprocess_input(image)
    preds = model.predict(image)
    return preds
```

Creating the basic view function and the upload and predict function.

```
@app.get('/')
def basic_view():
    return {"WELCOME": "GO TO /docs route, or /post or send post request to /predict "}}

@app.post('/predict')
def upload(file:UploadFile=File(...)):
    basepath = os.path.dirname(__file__)
    file_path = os.path.join(
        basepath, 'uploads', secure_filename(file.filename))

    preds = model_predict([file_path])
    if (preds<0):
        result="It's a strawberry!"
    else:
        result="That's not a strawberry!"
    return{ 'result': result }
```

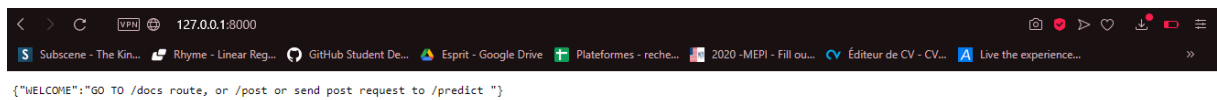
This line is necessary for running the app.

```
if __name__ == '__main__':
    uvicorn.run(app, hosts='127.0.0.1', port=8000)
```

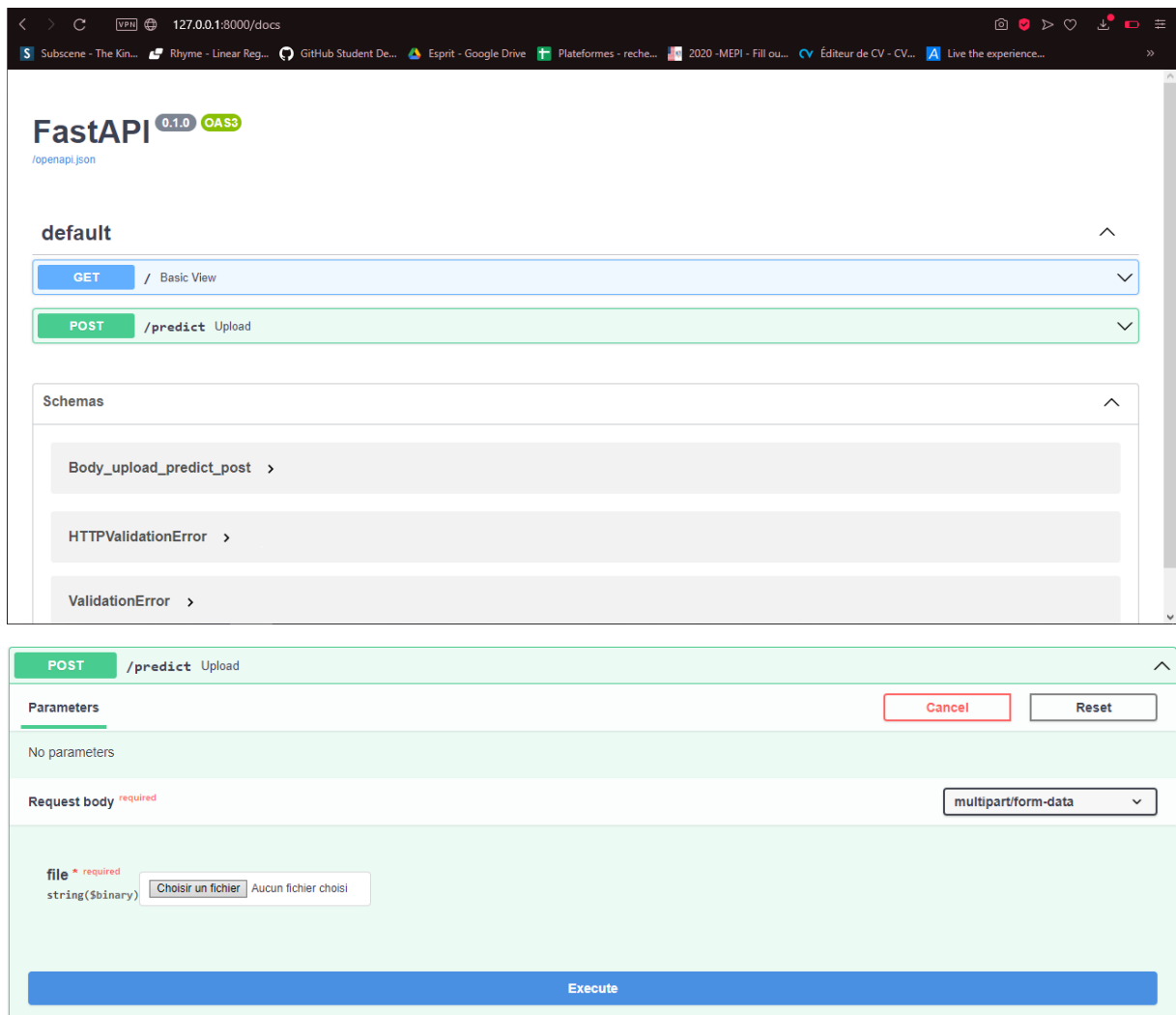
After debugging, i will be testing my app.

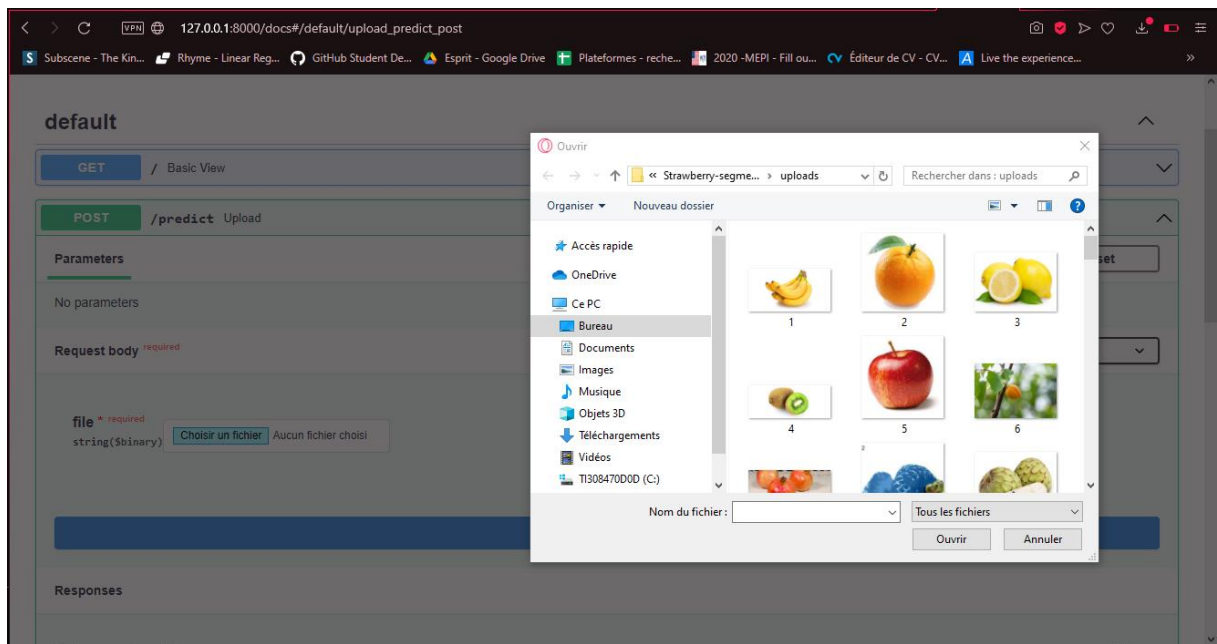
```
(venv) PS C:\Users\Ahmed\Desktop\Strawberry-segmentation - Copie> uvicorn app:app --reload
INFO:      Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:      Started reloader process [33144] using statreload
2021-07-18 02:36:21.838992: W tensorflow/stream_executor/platform/default/dso_loader.cc:60] Could not load dynamic library 'cudart64_110.dll'; dlderror:
cudart64_110.dll not found
2021-07-18 02:36:21.848363: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your ma
chine.
WARNING:tensorflow:SavedModel saved prior to TF 2.5 detected when loading Keras model. Please ensure that you are saving the model with model.save() or
tf.keras.models.save_model(), *NOT* tf.saved_model.save(). To confirm, there should be a file named "keras_metadata.pb" in the SavedModel directory.
2021-07-18 02:36:27.361074: W tensorflow/stream_executor/platform/default/dso_loader.cc:60] Could not load dynamic library 'nvcuda.dll'; dlderror: nvcud
a.dll not found
2021-07-18 02:36:27.368034: W tensorflow/stream_executor/cuda/cuda_driver.cc:326] failed call to cuInit: UNKNOWN ERROR (303)
2021-07-18 02:36:27.377967: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: NADA
2021-07-18 02:36:27.385713: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: NADA
2021-07-18 02:36:27.391597: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Li
brary (oneDNN) to use the following CPU instructions in performance-critical operations: AVX
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Model loaded. Check http://127.0.0.1:5000/
INFO:      Started server process [8780]
INFO:      Waiting for application startup.
INFO:      Application startup complete.
```

127.0.0.1/8000 on my browser to see what the API looks like.



127.0.0.1/8000/docs to test my functions.





Responses

Curl

```
curl -X 'POST' \
  'http://127.0.0.1:8000/predict' \
  -H 'accept: application/json' \
  -H 'Content-Type: multipart/form-data' \
  -F 'file=@5.jpg;type=image/jpeg'
```

Request URL

```
http://127.0.0.1:8000/predict
```

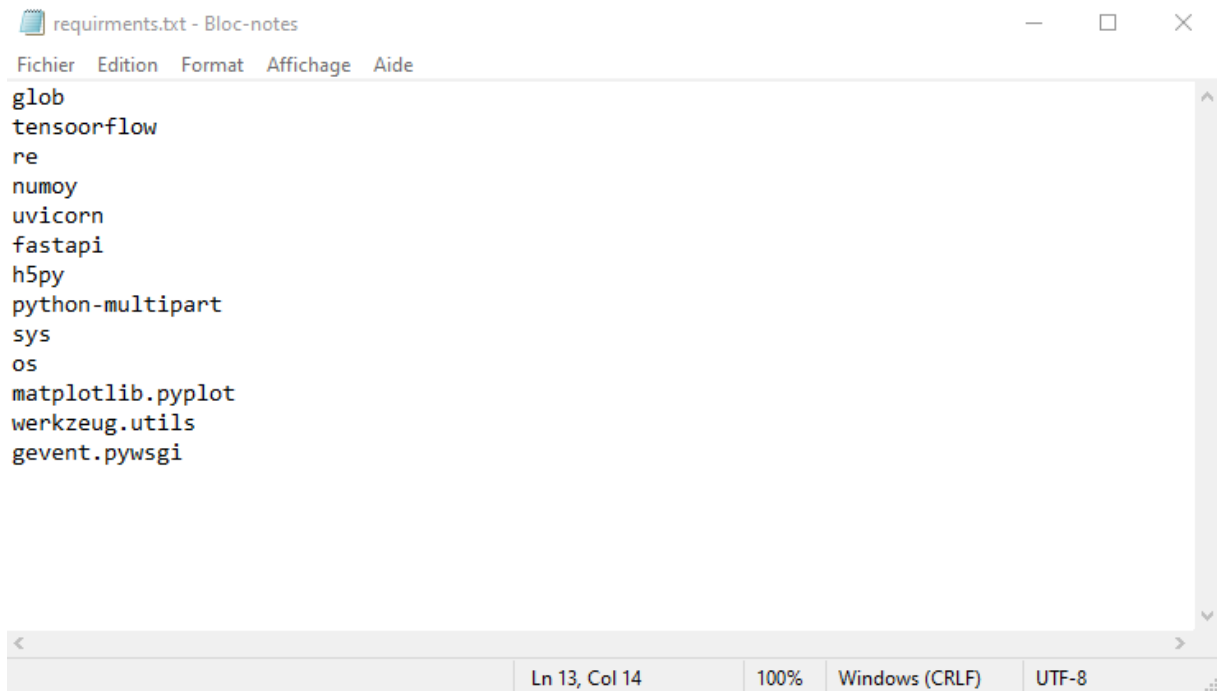
Server response

Code	Details
200	<p>Response body</p> <pre>{ "result": "That's not a strawberry!" }</pre> <p>Download</p>

requirements.txt is important for deployment on Heroku.

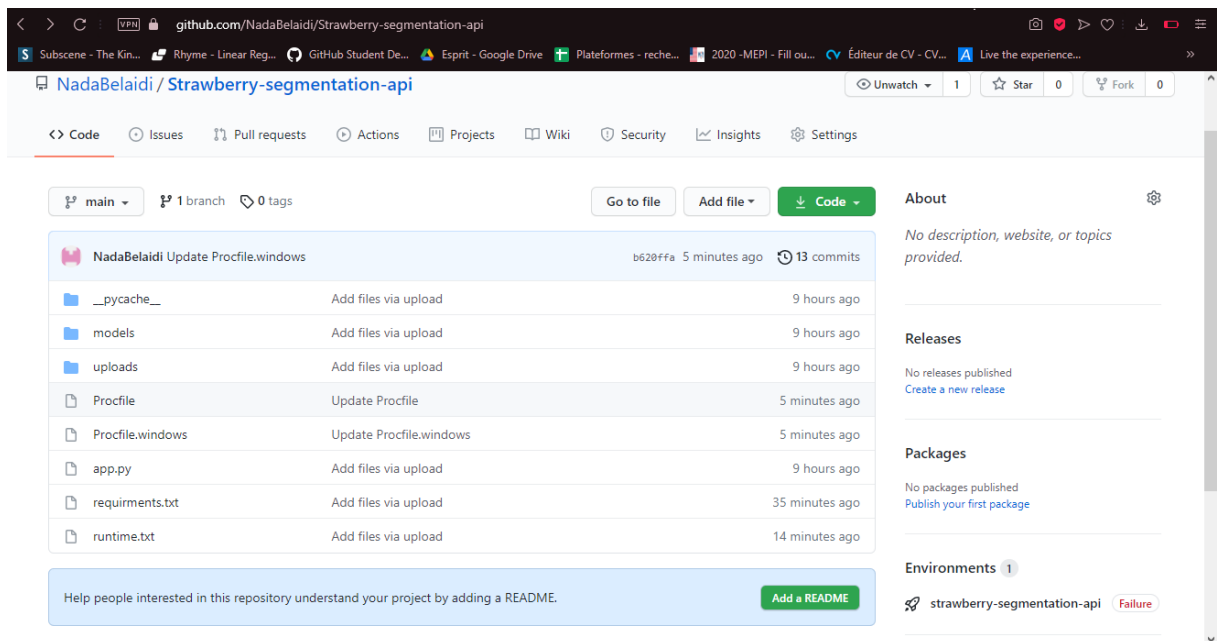
runtime.txt has python-3.9.6 in it.

And I also created the Procfile and the Procfile.windows



```
glob
tensorflow
re
numoy
uvicorn
fastapi
h5py
python-multipart
sys
os
matplotlib.pyplot
werkzeug.utils
gevent.pywsgi
```

I pushed the folder containing the API to my Github repository.



github.com/NadaBelaIdi/Strawberry-segmentation-api

NadaBelaIdi / Strawberry-segmentation-api

Unwatch 1 Star 0 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

File	Action	Time
NadaBelaIdi Update Procfile.windows	b620ffa	5 minutes ago 13 commits
__pycache__	Add files via upload	9 hours ago
models	Add files via upload	9 hours ago
uploads	Add files via upload	9 hours ago
Procfile	Update Procfile	5 minutes ago
Procfile.windows	Update Procfile.windows	5 minutes ago
app.py	Add files via upload	9 hours ago
requirments.txt	Add files via upload	35 minutes ago
runtime.txt	Add files via upload	14 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README

About No description, website, or topics provided.

Releases No releases published Create a new release

Packages No packages published Publish your first package

Environments 1

strawberry-segmentation-api Failure

I created an app on Heroku.

Salesforce Platform

HEROKU

Jump to Favorites, Apps, Pipelines, Spaces...

Create New App

App name

strawberry-segmentation-api

strawberry-segmentation-api is available

Choose a region

United States

Add to pipeline...

Create app

heroku.com Blogs Careers Documentation Support

Terms of Service Privacy Cookies © 2021 Salesforce.com

I connected it to my Github repository.

Connected to [NadaBelaidi/Strawberry-segmentation-api](#) by [NadaBelaidi](#) [Disconnect...](#)

Releases in the [activity feed](#) link to GitHub to view commit diffs

I will be now building it.

Build **main** [e7ced930](#) [View build log](#)

There was an issue deploying your app. View the **build log** for details.

```
tensorboard-2.5.0 tensorboard-data-server-0.6.1 tensorboard-plugin-wit-1.8.0 tensorflow-2.5.0rc0 termcolor-1.1.0 tf-estimator-nightly-2.5.0.dev2021032501 typing-extensions-3.7.4.3 urllib3-1.26.6 uvicorn-0.14.0 werkzeug-2.0.1 wrapt-1.12.1
----> Discovering process types
Procfile declares types -> web
----> Compressing...
!   Compiled slug size: 588.1M is too large (max is 500M).
!   See: http://devcenter.heroku.com/articles/slug-size
!   Push failed
```

Build finished

Release phase [View build log](#)

Deploy to Heroku

Building failed because I am using a free account, and my app was too large.

I did delete all unnecessary modules and libraries.

I believe that there is no solution for this, except for changing the whole app or going premium.

Same happened for the web app since it had the same deep learning model.

I will be trying to make another app for the assignment, meanwhile I will be submitting this in case it gets graded before I do.