

EECS 581/582 Final Project

Team 18

Date - March 26th, 2023

Sprint-11

ML integration part progress

60.5	Setup restAPI and flask	5	2	11
------	-------------------------	---	---	----

- Bikash was able to install restAPI dependencies needed and test it but is facing some problems with installing flask.

The screenshot shows a VS Code editor with a file named `Application.py` open. The code defines a simple Flask application with a single route `index()` that returns the string `'API Test for 582'`. The terminal at the bottom shows the command `python -m flask run` being executed, which starts the development server on `http://127.0.0.1:5000`. The terminal output includes several warnings about deprecated `FLASK_ENV` and `FLASK_DEBUG` variables, and a message indicating that the debugger is active. A browser window is also visible in the background, showing the URL `127.0.0.1:5000` and the response `API Test for 582`.

```
stackApi.py requirements.txt Application.py x
Application.py > ...
27
28 @app.route('/')
29 def index():
30     return 'API Test for 582'

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI se
rver instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
'FLASK_ENV' is deprecated and will not be used in Flask 2.3. Use 'FLASK_DEBUG' instead.
'FLASK_ENV' is deprecated and will not be used in Flask 2.3. Use 'FLASK_DEBUG' instead.
'FLASK_ENV' is deprecated and will not be used in Flask 2.3. Use 'FLASK_DEBUG' instead.
* Debugger is active!
* Debugger PIN: 823-273-206
127.0.0.1 - - [26/Mar/2023 21:20:05] "GET / HTTP/1.1" 200 -
```

60.3	Create notebook on AWS Sagemaker instance to load the yolov5 model - having issues with loading the model (partially completed - picture in the artifact)	8	8	11
60.3.1	Write inference.py model (AWS)	8	8	11

- Rudro is working on loading up our YOLOv5 model on to sagemaker, as well as writing the inference.py

```

1 import os
2 import glob as glob
3 import matplotlib.pyplot as plt
4 import requests
5 import numpy as np
6 import pandas as pd
7 from sagemaker.pytorch import PyTorch
8
9 def model_fn(model_dir):
10     device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
11     logger.info('Loading the model.')
12     # model = models.resnet50(pretrained=False)
13     # fc_inputs = model.fc.in_features
14
15     with open(os.path.join(model_dir, 'best_100.pt'), 'rb') as f:
16         model.load_state_dict(torch.load(f))
17     model.to(device).eval()
18     logger.info('Done loading model')
19     return model
20
21 def input_fn(request_body, request_content_type):
22     logger.info('Deserializing the input data.')
23     if content_type == 'application/json':
24         input_data = json.loads(request_body)
25         url = input_data['url']
26         logger.info(f'Image url: {url}')
27         image_data = Image.open(requests.get(url, stream=True).raw)
28
29         image_transform = transforms.Compose([
30             transforms.Resize(size=386),
31             transforms.CenterCrop(size=224),
32             transforms.ToTensor(),
33             transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
34         ])
35
36         return image_transform(image_data)
37     raise Exception(f'Requested unsupported Content Type in content_type {content_type}')
38
39 def predict_fn(input_data, model):
40     logger.info('Generating prediction based on input parameters.')
41     if torch.cuda.is_available():
42         input_data = input_data.view(1, 3, 224, 224).cuda()
43     else:
44         input_data = input_data.view(1, 3, 224, 224)

```

Fig: inference code script for the model

* At the end, we will choose between AWS or flask depending on which one is more appropriate for our project and which one we are able to run smoothly with our code.

```

22     logger.info('Deserializing the input data.')
23     if content_type == 'application/json':
24         input_data = json.loads(request_body)
25         url = input_data['url']
26         logger.info(f'Image url: {url}')
27         image_data = Image.open(requests.get(url, stream=True).raw)
28
29         image_transform = transforms.Compose([
30             transforms.Resize(size=256),
31             transforms.CenterCrop(size=224),
32             transforms.ToTensor(),
33             transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
34         ])
35
36         return image_transform(image_data)
37     raise Exception(f'Requested unsupported ContentType in content_type {content_type}')
38
39 def predict_fn(input_data, model):
40     logger.info('Generating prediction based on input parameters.')
41     if torch.cuda.is_available():
42         input_data = input_data.view(1, 3, 224, 224).cuda()
43     else:
44         input_data = input_data.view(1, 3, 224, 224)
45     with torch.no_grad():
46         model.eval()
47         out = model(input_data)
48         ps = torch.exp(out)
49     return ps
50
51 def output_fn(prediction, content_type):
52     logger.info('Serializing the generated output.')
53     classes = {0: 'Front', 1: 'Back'}
54
55     topk, topclass = prediction_output.topk(3, dim=1)
56     result = []
57
58     for i in range(3):
59         pred = {'prediction': classes[topclass.cpu().numpy()[0][i]], 'score': f'{topk.cpu().numpy()[0][i] * 100}%'}
60         logger.info(f'Adding prediction: {pred}')
61         result.append(pred)
62
63     if accept == 'application/json':
64         return json.dumps(result), accept
65     raise Exception(f'Requested unsupported ContentType in Accept:{accept}')

```

Fig: inference script continuation

62	Make a skeleton for project presentation	5	1	12
----	--	---	---	----

Planned the structure of the slides. Couldn't fully make the slides due to the uncertainty about the integration. We anticipate this to be done by the mid of the week 3/27.

Req 63:

Below is the implementation of the final screen that shows the total count. Count is made up for now as we are working on the integration.

12:05

79%

← Welcome

Wescoc Hall

Available Spaces: 45/436

Timings:

Mon-Thur: 9AM-5AM

Go to Menu



Req 49:

We are not doing it anymore. It is not useful

