App.py:

Import tensorflow

From flask import Flask,flash, request, render\_template

Import csv

Import math

Import os

Import numpy as np

From tensorflow.keras.preprocessing import image

From tensorflow.python.keras.models import load\_model

From werkzeug.utils import secure\_filename

Import tensorflow as tf

From tensorflow.keras.layers import BatchNormalization

Tmpl\_dir = os.path.join(os.path.dirname(os.path.abspath(\_\_file\_\_)), ‘templates’)

App = Flask(\_\_name\_\_, template\_folder=tmpl\_dir)

UPLOAD\_FOLDER = ‘static/uploads’

App.config[‘UPLOAD\_FOLDER’] = UPLOAD\_FOLDER

# define label meaning

Label = [‘apple pie:Estimate Calories 237 For 100 Grams Quantity’,

‘baby back ribs:Estimate Calories 276 For 100 Grams Quantity’,

‘baklava:Estimate Calories 403 For 100 Grams Quantity’,

‘beef carpaccio:Estimate Calories 231 For 100 Grams Quantity’,

‘beef tartare:Estimate Calories 246 For 100 Grams Quantity’,

‘beet salad:Estimate Calories 231 For 100 Grams Quantity’,

‘beignets:Estimate Calories 291 For 100 Grams Quantity’,

‘bibimbap:Estimate Calories 113 For 100 Grams Quantity’,

‘bread pudding:Estimate Calories 188 For 100 Grams Quantity’,

‘breakfast burrito:Estimate Calories 169 For 100 Grams Quantity’,

‘bruschetta:Estimate Calories 206 For 100 Grams Quantity’,

‘caesar salad:Estimate Calories 158 For 100 Grams Quantity’,

‘cannoli:Estimate Calories 293 For 100 Grams Quantity’,

‘caprese salad:Estimate Calories 136 For 100 Grams Quantity’,

‘carrot cake:Estimate Calories 333 For 100 Grams Quantity’,

‘ceviche:Estimate Calories 68 For 100 Grams Quantity’,

‘cheese plate:Estimate Calories 389 For 100 Grams Quantity’,

‘cheesecake:Estimate Calories 321 For 100 Grams Quantity’,

‘chicken curry:Estimate Calories 104 For 100 Grams Quantity’,

‘chicken quesadilla:Estimate Calories 216 For 100 Grams Quantity’,

‘chicken wings:Estimate Calories 328 For 100 Grams Quantity’,

‘chocolate cake:Estimate Calories 389 For 100 Grams Quantity’,

‘chocolate mousse:Estimate Calories 225 For 100 Grams Quantity’,

‘churros:Estimate Calories 396 For 100 Grams Quantity’,

‘clam chowder:Estimate Calories 79 For 100 Grams Quantity’,

‘club sandwich:Estimate Calories 234 For 100 Grams Quantity’,

‘crab cakes:Estimate Calories 173 For 100 Grams Quantity’,

‘crème brulee:Estimate Calories 343 For 100 Grams Quantity’,

‘croque madame:Estimate Calories 199 For 100 Grams Quantity’,

‘cup cakes:Estimate Calories 389 For 100 Grams Quantity’,

‘deviled eggs:Estimate Calories 225 For 100 Grams Quantity’,

‘donuts:Estimate Calories 421 For 100 Grams Quantity’,

‘dumplings:Estimate Calories 230 For 100 Grams Quantity’,

‘edamame:Estimate Calories 121 For 100 Grams Quantity’,

‘eggs benedict:Estimate Calories 260 For 100 Grams Quantity’,

‘escargots:Estimate Calories 90 For 100 Grams Quantity’,

‘falafel:Estimate Calories 416 For 100 Grams Quantity’,

‘filet mignon:Estimate Calories 267 For 100 Grams Quantity’,

‘fish and\_chips:Estimate Calories 134 For 100 Grams Quantity’,

‘foie gras:Estimate Calories 462 For 100 Grams Quantity’,

‘french fries:Estimate Calories 312 For 100 Grams Quantity’,

‘french onion soup:Estimate Calories 137 For 100 Grams Quantity’,

‘french toast:Estimate Calories 263 For 100 Grams Quantity’,

‘fried calamari:Estimate Calories 249 For 100 Grams Quantity’,

‘fried rice:Estimate Calories 174 For 100 Grams Quantity’,

‘frozen yogurt:Estimate Calories 127 For 100 Grams Quantity’,

‘garlic bread:Estimate Calories 350 For 100 Grams Quantity’,

‘gnocchi:Estimate Calories 201 For 100 Grams Quantity’,

‘greek salad:Estimate Calories 113 For 100 Grams Quantity’,

‘grilled cheese sandwich:Estimate Calories 344 For 100 Grams Quantity’,

‘grilled salmon:Estimate Calories 206 For 100 Grams Quantity’,

‘guacamole:Estimate Calories 151 For 100 Grams Quantity’,

‘gyoza:Estimate Calories 211 For 100 Grams Quantity’,

‘hamburger:Estimate Calories 239 For 100 Grams Quantity’,

‘hot and sour soup:Estimate Calories 39 For 100 Grams Quantity’,

‘hot dog:Estimate Calories 322 For 100 Grams Quantity’,

‘huevos rancheros:Estimate Calories 143 For 100 Grams Quantity’,

‘hummus:Estimate Calories 166 For 100 Grams Quantity’,

‘ice cream:Estimate Calories 207 For 100 Grams Quantity’,

‘lasagna:Estimate Calories 156 For 100 Grams Quantity’,

‘lobster bisque:Estimate Calories 106 For 100 Grams Quantity’,

‘lobster roll sandwich:Estimate Calories 199 For 100 Grams Quantity’,

‘macaroni and cheese:Estimate Calories 190 For 100 Grams Quantity’,

‘macarons:Estimate Calories 384 For 100 Grams Quantity’,

‘miso soup:Estimate Calories 24 For 100 Grams Quantity’,

‘mussels:Estimate Calories 172 For 100 Grams Quantity’,

‘nachos:Estimate Calories 224 For 100 Grams Quantity’,

‘omelette:Estimate Calories 181 For 100 Grams Quantity’,

‘onion rings:Estimate Calories 356 For 100 Grams Quantity’,

‘oysters:Estimate Calories 163 For 100 Grams Quantity’,

‘pad thai:Estimate Calories 170 For 100 Grams Quantity’,

‘paella:Estimate Calories 183 For 100 Grams Quantity’,

‘pancakes:Estimate Calories 227 For 100 Grams Quantity’,

‘panna cotta:Estimate Calories 319 For 100 Grams Quantity’,

‘peking duck:Estimate Calories 241 For 100 Grams Quantity’,

‘pho:Estimate Calories 90 For 100 Grams Quantity’,

‘pizza:Estimate Calories 266 For 100 Grams Quantity’,

‘pork chop:Estimate Calories 209 For 100 Grams Quantity’,

‘poutine:Estimate Calories 222 For 100 Grams Quantity’,

‘prime rib:Estimate Calories 341 For 100 Grams Quantity’,

‘pulled pork sandwich:Estimate Calories 175 For 100 Grams Quantity’,

‘ramen:Estimate Calories 135 For 100 Grams Quantity’,

‘ravioli:Estimate Calories 179 For 100 Grams Quantity’,

‘red velvet cake:Estimate Calories 337 For 100 Grams Quantity’,

‘risotto:Estimate Calories 122 For 100 Grams Quantity’,

‘samosa:Estimate Calories 261 For 100 Grams Quantity’,

‘sashimi:Estimate Calories 124 For 100 Grams Quantity’,

‘scallops:Estimate Calories 111 For 100 Grams Quantity’,

‘seaweed salad:Estimate Calories 115 For 100 Grams Quantity’,

‘shrimp and grits:Estimate Calories 149 For 100 Grams Quantity’,

‘spaghetti bolognese:Estimate Calories 101 For 100 Grams Quantity’,

‘spaghetti carbonara:Estimate Calories 199 For 100 Grams Quantity’,

‘spring rolls:Estimate Calories 230 For 100 Grams Quantity’,

‘steak:Estimate Calories 278 For 100 Grams Quantity’,

‘strawberry shortcake:Estimate Calories 172 For 100 Grams Quantity’,

‘sushi:Estimate Calories 165 For 100 Grams Quantity’,

‘tacos:Estimate Calories 206 For 100 Grams Quantity’,

‘takoyaki:Estimate Calories 149 For 100 Grams Quantity’,

‘tiramisu:Estimate Calories 329 For 100 Grams Quantity’,

‘tuna tartare:Estimate Calories 176 For 100 Grams Quantity’,

‘waffles:Estimate Calories 291 For 100 Grams Quantity’]

Nu\_link = ‘https://www.nutritionix.com/food/’

# Loading the best saved model to make predictions.

Tensorflow.keras.backend.clear\_session()

Model = tf.keras.models.load\_model(‘food.h5’)

Print(‘model successfully loaded!’)

Start = [0]

Passed = [0]

Pack = [[]]

Num = [0]

Nutrients = [

{‘name’: ‘protein’, ‘value’: 0.0},

{‘name’: ‘calcium’, ‘value’: 0.0},

{‘name’: ‘fat’, ‘value’: 0.0},

{‘name’: ‘carbohydrates’, ‘value’: 0.0},

{‘name’: ‘vitamins’, ‘value’: 0.0}

]

With open(‘nutrition101.csv’, ‘r’) as file:

Reader = csv.reader(file)

Nutrition\_table = dict()

For I, row in enumerate(reader):

If I == 0:

Name = ‘’

Continue

Else:

Name = row[1].strip()

Nutrition\_table[name] = [

{‘name’: ‘protein’, ‘value’: float(row[2])},

{‘name’: ‘calcium’, ‘value’: float(row[3])},

{‘name’: ‘fat’, ‘value’: float(row[4])},

{‘name’: ‘carbohydrates’, ‘value’: float(row[5])},

{‘name’: ‘vitamins’, ‘value’: float(row[6])}

]

@app.route(“/”)

@app.route(“/index”)

Def index():

Return render\_template(‘index.html’)

@app.route(“/login”)

Def login():

Return render\_template(‘login.html’)

@app.route(“/chart”)

Def chart():

Return render\_template(‘chart.html’)

@app.route(‘/recognize’)

Def recognize():

Return render\_template(‘recognize.html’)

@app.route(‘/upload’, methods=[‘POST’])

Def upload():

File = request.files.getlist(“img”)

For f in file:

Filename = secure\_filename(str(num[0] + 500) + ‘.jpg’)

Num[0] += 1

Name = os.path.join(app.config[‘UPLOAD\_FOLDER’], filename)

Print(‘save name’, name)

f.save(name)

pack[0] = []

return render\_template(‘recognize.html’, img=file)

@app.route(‘/predict’)

Def predict():

Result = []

# pack = []

Print(‘total image’, num[0])

For I in range(start[0], num[0]):

Pa = dict()

Filename = f’{UPLOAD\_FOLDER}/{I + 500}.jpg’

Print(‘image filepath’, filename)

Pred\_img = filename

Pred\_img = image.load\_img(pred\_img, target\_size=(128, 128))

Pred\_img = image.img\_to\_array(pred\_img)

Pred\_img = np.expand\_dims(pred\_img, axis=0)

Pred\_img = pred\_img / 255.

Pred = model.predict(pred\_img)

Print(“Pred”)

Print(pred)

If math.isnan(pred[0][0]) and math.isnan(pred[0][1]) and \

Math.isnan(pred[0][2]) and math.isnan(pred[0][3]):

Pred = np.array([0.05, 0.05, 0.05, 0.07, 0.09, 0.19, 0.55, 0.0, 0.0, 0.0, 0.0])

Top = pred.argsort()[0][-3:]

Label.sort()

\_true = label[top[2]]

\_trues = label[top[2]]

Print(\_trues)

Pa[‘image’] = f’{UPLOAD\_FOLDER}/{I + 500}.jpg’

X = dict()

X[\_true] = float(“{:.2f}”.format(pred[0][top[2]] \* 100))

Print(x[\_true])

X[label[top[1]]] = float(“{:.2f}”.format(pred[0][top[1]] \* 100))

Print(x[label[top[1]]])

X[label[top[0]]] = float(“{:.2f}”.format(pred[0][top[0]] \* 100))

Pa[‘result’] = x

Print(x)

Pa[‘nutrition’] = nutrition\_table[\_true]

Pa[‘food’] = f’{nu\_link}{\_true}’

Pa[‘idx’] = I – start[0]

Pa[‘quantity’] = 100

Pack[0].append(pa)

Passed[0] += 1

Start[0] = passed[0]

Print(‘successfully packed’)

# compute the average source of calories

For p in pack[0]:

Nutrients[0][‘value’] = (nutrients[0][‘value’] + p[‘nutrition’][0][‘value’])

Nutrients[1][‘value’] = (nutrients[1][‘value’] + p[‘nutrition’][1][‘value’])

Nutrients[2][‘value’] = (nutrients[2][‘value’] + p[‘nutrition’][2][‘value’])

Nutrients[3][‘value’] = (nutrients[3][‘value’] + p[‘nutrition’][3][‘value’])

Nutrients[4][‘value’] = (nutrients[4][‘value’] + p[‘nutrition’][4][‘value’])

Nutrients[0][‘value’] = nutrients[0][‘value’] / num[0]

Nutrients[1][‘value’] = nutrients[1][‘value’] / num[0]

Nutrients[2][‘value’] = nutrients[2][‘value’] / num[0]

Nutrients[3][‘value’] = nutrients[3][‘value’] / num[0]

Nutrients[4][‘value’] = nutrients[4][‘value’] / num[0]

Return render\_template(‘results.html’, pack=pack[0], whole\_nutrition=nutrients, prediction = \_trues)

@app.route(‘/update’, methods=[‘POST’])

Def update():

Return render\_template(‘index.html’, img=’static/P2.jpg’)

If \_\_name\_\_ == “\_\_main\_\_”:

Import click

@click.command()

@click.option(‘—debug’, is\_flag=True)

@click.option(‘—threaded’, is\_flag=True)

@click.argument(‘HOST’, default=’127.0.0.1’)

@click.argument(‘PORT’, default=5000, type=int)

Def run(debug, threaded, host, port):

“””

This function handles command line parameters.

Run the server using

Python server.py

Show the help text using

Python server.py –help

“””

HOST, PORT = host, port

App.run(host=HOST, port=PORT, debug=debug, threaded=threaded)

Run()

Get\_nutrition\_data.py:

Import requests

Import pandas as pd

Def get\_nutrition(food\_name):

Nutrition\_data = pd.DataFrame(columns=[‘name’, ‘protein’, ‘calcium’, ‘fat’, ‘carbohydrates’, ‘vitamins’])

For name in food\_name:

url = <https://api.nal.usda.gov/fdc/v1/foods/search?api_key=d4D6dSOc81pTAOY2gsNZ0YhjkMlhStLJRoII5SJu&query=> + name

response = requests.get(url)

data = response.json()

flatten\_json = pd.json\_normalize(data[“foods”])

first\_food = flatten\_json.iloc[0]

first\_food\_nutrition\_list = first\_food.foodNutrients

for item in first\_food\_nutrition\_list:

if item[‘nutrientNumber’] == “203”:

protein = item[‘value’]

continue

if item[‘nutrientNumber’] == “301”:

calcium = item[‘value’]

continue

if item[‘nutrientNumber’] == “204”:

fat = item[‘value’]

continue

if item[‘nutrientNumber’] == “205”:

carbs = item[‘value’]

continue

if item[‘nutrientNumber’] == “318”:

vitamin\_a = item[‘value’]

continue

if item[‘nutrientNumber’] == “401”:

vitamin\_c = item[‘value’]

continue

vitamins = float(vitamin\_a) + float(vitamin\_c)

print(name)

nutrition\_data = nutrition\_data.append({

‘name’: name,

‘protein’: protein,

‘calcium’: calcium / 1000,

‘fat’: fat,

‘carbohydrates’: carbs,

‘vitamins’: vitamins / 1000

}, ignore\_index=True)

Return nutrition\_data

Nutrition101 = get\_nutrition([‘apple pie:Estimate Calories 237 For 100 Grams Quantity’,

‘baby back ribs:Estimate Calories 276 For 100 Grams Quantity’,

‘baklava:Estimate Calories 403 For 100 Grams Quantity’,

‘beef carpaccio:Estimate Calories 231 For 100 Grams Quantity’,

‘beef tartare:Estimate Calories 246 For 100 Grams Quantity’,

‘beet salad:Estimate Calories 231 For 100 Grams Quantity’,

‘beignets:Estimate Calories 291 For 100 Grams Quantity’,

‘bibimbap:Estimate Calories 113 For 100 Grams Quantity’,

‘bread pudding:Estimate Calories 188 For 100 Grams Quantity’,

‘breakfast burrito:Estimate Calories 169 For 100 Grams Quantity’,

‘bruschetta:Estimate Calories 206 For 100 Grams Quantity’,

‘caesar salad:Estimate Calories 158 For 100 Grams Quantity’,

‘cannoli:Estimate Calories 293 For 100 Grams Quantity’,

‘caprese salad:Estimate Calories 136 For 100 Grams Quantity’,

‘carrot cake:Estimate Calories 333 For 100 Grams Quantity’,

‘ceviche:Estimate Calories 68 For 100 Grams Quantity’,

‘cheese plate:Estimate Calories 389 For 100 Grams Quantity’,

‘cheesecake:Estimate Calories 321 For 100 Grams Quantity’,

‘chicken curry:Estimate Calories 104 For 100 Grams Quantity’,

‘chicken quesadilla:Estimate Calories 216 For 100 Grams Quantity’,

‘chicken wings:Estimate Calories 328 For 100 Grams Quantity’,

‘chocolate cake:Estimate Calories 389 For 100 Grams Quantity’,

‘chocolate mousse:Estimate Calories 225 For 100 Grams Quantity’,

‘churros:Estimate Calories 396 For 100 Grams Quantity’,

‘clam chowder:Estimate Calories 79 For 100 Grams Quantity’,

‘club sandwich:Estimate Calories 234 For 100 Grams Quantity’,

‘crab cakes:Estimate Calories 173 For 100 Grams Quantity’,

‘crème brulee:Estimate Calories 343 For 100 Grams Quantity’,

‘croque madame:Estimate Calories 199 For 100 Grams Quantity’,

‘cup cakes:Estimate Calories 389 For 100 Grams Quantity’,

‘deviled eggs:Estimate Calories 225 For 100 Grams Quantity’,

‘donuts:Estimate Calories 421 For 100 Grams Quantity’,

‘dumplings:Estimate Calories 230 For 100 Grams Quantity’,

‘edamame:Estimate Calories 121 For 100 Grams Quantity’,

‘eggs benedict:Estimate Calories 260 For 100 Grams Quantity’,

‘escargots:Estimate Calories 90 For 100 Grams Quantity’,

‘falafel:Estimate Calories 416 For 100 Grams Quantity’,

‘filet mignon:Estimate Calories 267 For 100 Grams Quantity’,

‘fish and\_chips:Estimate Calories 134 For 100 Grams Quantity’,

‘foie gras:Estimate Calories 462 For 100 Grams Quantity’,

‘french fries:Estimate Calories 312 For 100 Grams Quantity’,

‘french onion soup:Estimate Calories 137 For 100 Grams Quantity’,

‘french toast:Estimate Calories 263 For 100 Grams Quantity’,

‘fried calamari:Estimate Calories 249 For 100 Grams Quantity’,

‘fried rice:Estimate Calories 174 For 100 Grams Quantity’,

‘frozen yogurt:Estimate Calories 127 For 100 Grams Quantity’,

‘garlic bread:Estimate Calories 350 For 100 Grams Quantity’,

‘gnocchi:Estimate Calories 201 For 100 Grams Quantity’,

‘greek salad:Estimate Calories 113 For 100 Grams Quantity’,

‘grilled cheese sandwich:Estimate Calories 344 For 100 Grams Quantity’,

‘grilled salmon:Estimate Calories 206 For 100 Grams Quantity’,

‘guacamole:Estimate Calories 151 For 100 Grams Quantity’,

‘gyoza:Estimate Calories 211 For 100 Grams Quantity’,

‘hamburger:Estimate Calories 239 For 100 Grams Quantity’,

‘hot and sour soup:Estimate Calories 39 For 100 Grams Quantity’,

‘hot dog:Estimate Calories 322 For 100 Grams Quantity’,

‘huevos rancheros:Estimate Calories 143 For 100 Grams Quantity’,

‘hummus:Estimate Calories 166 For 100 Grams Quantity’,

‘ice cream:Estimate Calories 207 For 100 Grams Quantity’,

‘lasagna:Estimate Calories 156 For 100 Grams Quantity’,

‘lobster bisque:Estimate Calories 106 For 100 Grams Quantity’,

‘lobster roll sandwich:Estimate Calories 199 For 100 Grams Quantity’,

‘macaroni and cheese:Estimate Calories 190 For 100 Grams Quantity’,

‘macarons:Estimate Calories 384 For 100 Grams Quantity’,

‘miso soup:Estimate Calories 24 For 100 Grams Quantity’,

‘mussels:Estimate Calories 172 For 100 Grams Quantity’,

‘nachos:Estimate Calories 224 For 100 Grams Quantity’,

‘omelette:Estimate Calories 181 For 100 Grams Quantity’,

‘onion rings:Estimate Calories 356 For 100 Grams Quantity’,

‘oysters:Estimate Calories 163 For 100 Grams Quantity’,

‘pad thai:Estimate Calories 170 For 100 Grams Quantity’,

‘paella:Estimate Calories 183 For 100 Grams Quantity’,

‘pancakes:Estimate Calories 227 For 100 Grams Quantity’,

‘panna cotta:Estimate Calories 319 For 100 Grams Quantity’,

‘peking duck:Estimate Calories 241 For 100 Grams Quantity’,

‘pho:Estimate Calories 90 For 100 Grams Quantity’,

‘pizza:Estimate Calories 266 For 100 Grams Quantity’,

‘pork chop:Estimate Calories 209 For 100 Grams Quantity’,

‘poutine:Estimate Calories 222 For 100 Grams Quantity’,

‘prime rib:Estimate Calories 341 For 100 Grams Quantity’,

‘pulled pork sandwich:Estimate Calories 175 For 100 Grams Quantity’,

‘ramen:Estimate Calories 135 For 100 Grams Quantity’,

‘ravioli:Estimate Calories 179 For 100 Grams Quantity’,

‘red velvet cake:Estimate Calories 337 For 100 Grams Quantity’,

‘risotto:Estimate Calories 122 For 100 Grams Quantity’,

‘samosa:Estimate Calories 261 For 100 Grams Quantity’,

‘sashimi:Estimate Calories 124 For 100 Grams Quantity’,

‘scallops:Estimate Calories 111 For 100 Grams Quantity’,

‘seaweed salad:Estimate Calories 115 For 100 Grams Quantity’,

‘shrimp and grits:Estimate Calories 149 For 100 Grams Quantity’,

‘spaghetti bolognese:Estimate Calories 101 For 100 Grams Quantity’,

‘spaghetti carbonara:Estimate Calories 199 For 100 Grams Quantity’,

‘spring rolls:Estimate Calories 230 For 100 Grams Quantity’,

‘steak:Estimate Calories 278 For 100 Grams Quantity’,

‘strawberry shortcake:Estimate Calories 172 For 100 Grams Quantity’,

‘sushi:Estimate Calories 165 For 100 Grams Quantity’,

‘tacos:Estimate Calories 206 For 100 Grams Quantity’,

‘takoyaki:Estimate Calories 149 For 100 Grams Quantity’,

‘tiramisu:Estimate Calories 329 For 100 Grams Quantity’,

‘tuna tartare:Estimate Calories 176 For 100 Grams Quantity’,

‘waffles:Estimate Calories 291 For 100 Grams Quantity’]

)

Nutrition101 = nutrition101.reset\_index(drop=True)

Nutrition101.to\_csv(“nutrition101.csv”)

-----------------------------------------------------------------------------