|  |
| --- |
| import os |
|  | import requests |
|  | from flask import Flask, request, render\_template, redirect, url\_for |
|  | from cloudant.client import Cloudant |
|  |  |
|  | import numpy as np |
|  | from tensorflow.keras.models import load\_model |
|  | from tensorflow.keras.preprocessing import image |
|  | from tensorflow.keras.applications.inception\_v3 import preprocess\_input |
|  |  |
|  |  |
|  | model = load\_model(r"model\Updated-xception-diabetic-retinopathy.h5") |
|  |  |
|  | app = Flask(\_\_name\_\_) |
|  |  |
|  |  |
|  |  |
|  | #Authenticate using an IAM API key |
|  | client = Cloudant.iam('55a4f815-9a4a-4711-b663-d2733b89f3f9-bluemix','Ga7SGlD639xERt-F6egdft3j2dNntgT5CelqppKEgSLp', connect=True) |
|  |  |
|  |  |
|  |  |
|  | #create a database using an initialized client |
|  | my\_database = client.create\_database('ibm-deeplearning') |
|  |  |
|  |  |
|  |  |
|  | # @app.route('/') |
|  | # def index(): |
|  | # return render\_template('index.html') |
|  |  |
|  |  |
|  |  |
|  | @app.route('/index') |
|  | def home(): |
|  | return render\_template('index.html') |
|  |  |
|  |  |
|  | @app.route('/') |
|  | def index(): |
|  | return render\_template('register.html') |
|  |  |
|  | #registration page |
|  | @app.route('/register') |
|  | def register(): |
|  | return render\_template('register.html') |
|  |  |
|  |  |
|  |  |
|  | @app.route('/afterreg', methods=['POST']) |
|  | def afterreg(): |
|  |  |
|  | name = request.form.get('name') |
|  | email = request.form.get('emailid') |
|  | password = request.form.get('pass') |
|  |  |
|  | print(name,email,password) |
|  |  |
|  | data = { |
|  | '\_id':email, |
|  | 'name':name, |
|  | 'psw':password, |
|  | } |
|  |  |
|  | print(data) |
|  |  |
|  | query = {'\_id': {'$eq': data['\_id']}} |
|  |  |
|  | docs = my\_database.get\_query\_result(query) |
|  |  |
|  | print(docs) |
|  |  |
|  | print(len(docs.all())) |
|  |  |
|  | if(len(docs.all())==0): |
|  | url = my\_database.create\_document(data) |
|  | return render\_template('register.html', pred="Registration successfull, Please login using your details") |
|  | else: |
|  | return render\_template('register.html', pred="You are already a member, Please login using your details") |
|  |  |
|  |  |
|  |  |
|  | #login page |
|  | @app.route('/login') |
|  | def login(): |
|  | return render\_template('login.html') |
|  |  |
|  |  |
|  |  |
|  | @app.route('/afterlogin', methods=['POST']) |
|  | def afterlogin(): |
|  |  |
|  | user = request.form.get('emailid') |
|  | passw = request.form.get('pass') |
|  | print(user,passw) |
|  |  |
|  | query = {'\_id': {'$eq': user}} |
|  |  |
|  | docs = my\_database.get\_query\_result(query) |
|  | print(docs) |
|  |  |
|  | print(len(docs.all())) |
|  | if(len(docs.all())==0): |
|  | return render\_template('login.html', pred="The username is not found, please Register") |
|  | else: |
|  | if((user==docs[0][0]['\_id'] and passw==docs[0][0]['psw'])): |
|  | return render\_template('index.html') |
|  | else: |
|  | print('Invalid User') |
|  |  |
|  |  |
|  | @app.route('/logout') |
|  | def logout(): |
|  | return render\_template('logout.html') |
|  |  |
|  |  |
|  | #prediction |
|  | @app.route('/prediction') |
|  | def prediction(): |
|  | return render\_template('prediction.html',pred=None) |
|  |  |
|  |  |
|  | @app.route('/predict', methods=['POST']) |
|  | def predict(): |
|  |  |
|  | f = request.files['image'] |
|  | basepath = os.path.dirname(\_\_file\_\_) #getting the current path i.e. where app.py is present |
|  | filepath=os.path.join(basepath,'uploads',f.filename) |
|  | f.save(filepath) |
|  |  |
|  | img = image.load\_img(filepath,target\_size=(299,299)) |
|  | x = image.img\_to\_array(img) |
|  | x=np.expand\_dims(x,axis=0)#used for adding one more dimension |
|  | img\_data=preprocess\_input(x) |
|  | prediction=np.argmax(model.predict(img\_data),axis=1) |
|  |  |
|  | print("prediction is", prediction) |
|  |  |
|  | index=['No Diabetic Retinopathy', 'Mild Diabetic Retinopathy', 'Moderate Diabetic Retinopathy', 'Severe Diabetic Retinopathy', 'Proliferative Diabetic Retinopathy'] |
|  |  |
|  | res = str(index[prediction[0]]) |
|  |  |
|  | return render\_template('prediction.html',pred=res) |
|  |  |
|  | if \_\_name\_\_ == "\_\_main\_\_": |
|  | app.run(debug=False) |