PROJECT REPORT

Project Name: DEEP LEARING FOR BREAST CANCER RISK PREDICTION USING PYTHON

Internship Project at SmartInternz
Category: Artificial Intelligence

Name: SIMRAN KAUR
Project Id: SPS_PRO_202
Internship title: RSIP Career Basic AI 058
Email Id: simransaini7575@gmail.com

Deep Learning Techniques for Breast Cancer Risk Prediction using Python

1 INTRODUCTION

1.1 Overview:

Breast cancer is a type of cancer which forms in the cells of the breasts. Breast cancer mainly occur in women and rarely in men. Breast cancer is a dominant cancer in women worldwide and is rapidly increasing in developing countries like Belgium , Luxembourg, Netherlands etcand where the most of cases are diagnosed in late stages. Breast cancer is the worlds second dangerous cancer after lung cancer. In 2018 year according to the imfomation provided by World Cancer Research Fund it is calculated that over 2 million new cases were found from which 626,679 deaths were estimated. From all the cancers, Breast cancer in new cancer cases constitutes of 11.6% and come up with 24.2% of cancers in women. Symptoms of breast cancer are a lump in the breast, bloody discharge from the nipple and changes in the shape or texture of the nipple or breast. Breast cancer is one of the main causes of cancer death worldwide. Hence, it is important to have a model which predicts if Breast cancer is benign or malignant.

1.2 Purpose:

Purpose of Breast cancer risk prediction model is to predict is a person has Breast cancer benign or malignant. Early prdiction significantly increases the chances of correct treatment and survival, but this process is tedious and often leads to a disagreement between pathologists. Computer-aided diagnosis systems showed potential for improving diagnostic accuracy. But early detection and prevention can significantly reduce the chances of death. Hence, It is important to detect breast cancer as early as possible. As the diagnosis of Breast cancer manually takes number of hours and the availability of systems are less, there is a big need to make an automatic diagnosis system for the early detection of cancer. Deep Learning techniques contribute a lot in the development of such system.

2 LITERATURE SURVEY

2.1 Existing problem:

Every year, death rate in woman increases drastically due to breast cancer. Existing problem in Breast caner diagnosis is that it takes large hours and is higly costly. Beacuse of higly costly most of the people can not afford the diagnosis of breast cancer. Along with that their is less availability of systems beacuse systems used for Breast caner diagnosis are highly expensive.Most of the time people identify they have Breast cancer at late stages hence their survival rate becomes less and where as early prdiction significantly increases the chances of correct treatment and survival.

2.2 Proposed solution:

In response to the problem mentioned, the aim to develop a Deep Learning Techniques for Breast Cancer Risk Prediction using Python. This model also uses flask in order to interact with user. In this proposed system we are able to identify the rate of breast cancer. By prediction of Breast cancer in healthcare we are able to take suitable measures accordingly to control the cancer tissue. Due to the large size of each image in the training dataset, we propose a technique which consists of two consecutive convolutional neural networks. By using deep neural networks we are predicting the cancer is benign or malignant. Users have feasibility to upload scanned images on a web page to know about the status.
The Dataset used for the making of this project is taken form kaggle and name of dataset is Breast Histopathology Images. After taking dataset make a folder Dataset and in that make to folder testset and trainset. Then in these forlder make new two new folder benign and malignant and then add images in both folders. Then do data preprocessing using ImageDataGenerator. Then start building model for that import libraries, inisialize the model, add CNN layers, add dense layers, configure the learning process, tain and test the model, optimize the model and at last save the model. The last step is to make a webpage using flask. For that make a html, css, js files and a app.py file. And run this app.py file in comand prompt and see the result in http://localhost:5000/

3 THEORITICAL ANALYSIS

3.1 Block Diagram:

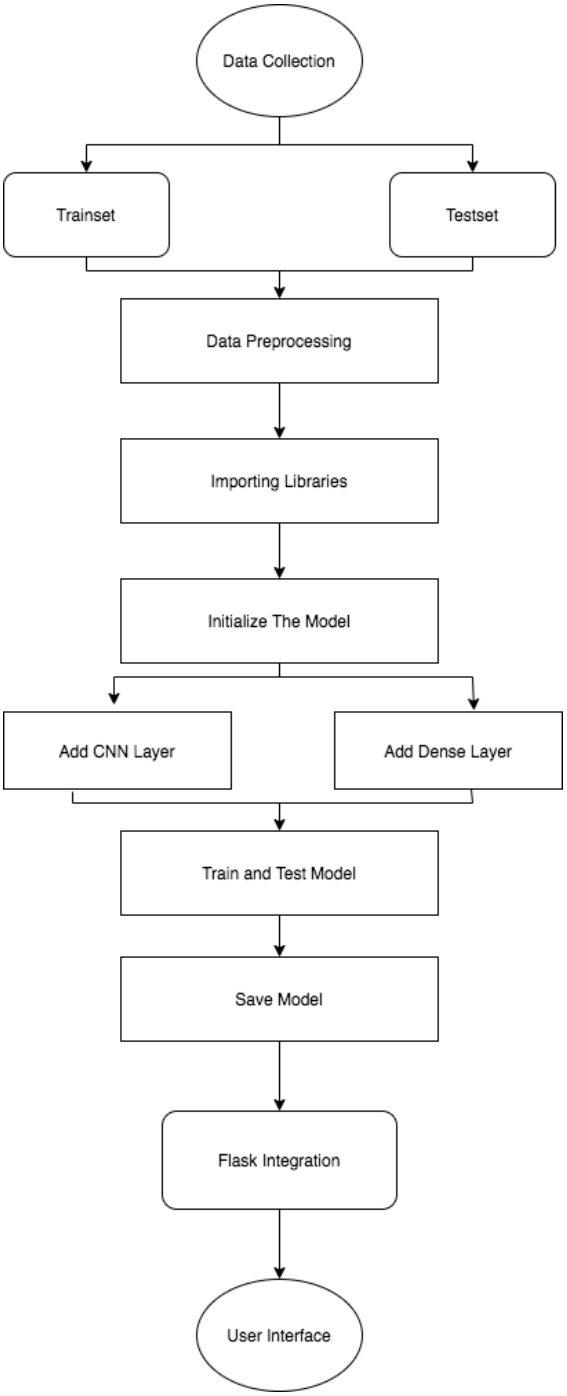


fig. 1 : Block Diagram

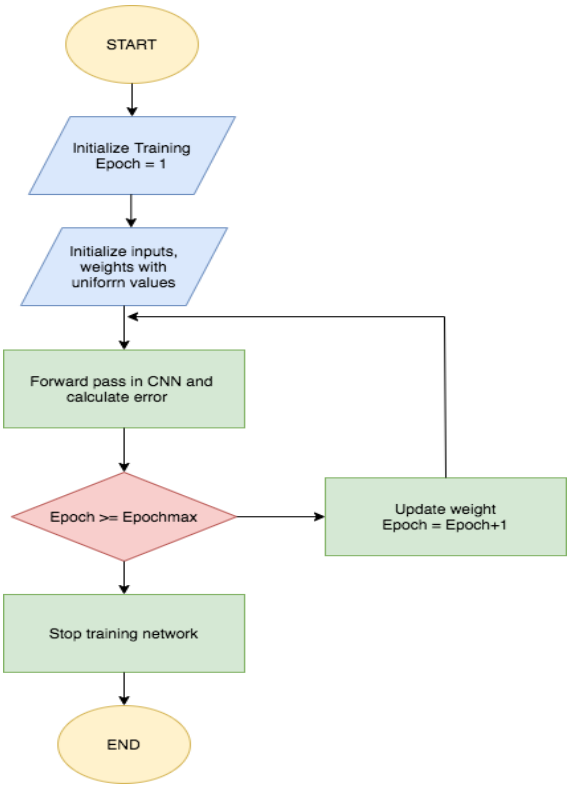
3.2 Hardware / Software designing:

Since the Breast Cancer Risk Prediction is a web application it does not have any designated hardware, Hence it does not have any direct hardware designing.
Python, Python Web Frame Works, CNN, Flask Integration are the tools and software used for the making of Breast Cancer Risk Prediction.

4 EXPERIMENTAL INVESTIGATIONS

Experimental Investigations plays a important role in Deep Learing. Breast Cancer Risk Prediction falls under the Experimental technique. After doing a detailed investigation on the techniques available for implementation of Breast Cancer Risk Prediction systems and going through many research papers written on the subject of Cancer Risk Prediction systems, we have come up with the proposed system. The proposed system aims to help the user by predicting if user has Breast cancer as benign or malignant.

5 FLOWCHART



6 RESULT

The Breast Cancer Risk Prediction was successfully made using the Python, Python Web Frame Works, CNN, Flask Integration Web Frame Works, CNN, Flask Integration.

7 ADVANTAGES & DISADVANTAGES

Advantages:

- The deep neural network shows a higher risk association for Breast Cancer as compare to the best mammographic density model. The false negative rate which is the rate at which women who were not categorized as high-risk were later diagnosed with breast cancer was lower for the deep neural network than for the best mammographic density model which is one of the advantage of deep learning models.Other than that deep neural network are better than density-based models also and it did not have the same bias as the density-based model. Its predictive accuracy was not negatively affected by more aggressive cancer subtypes.
- Other advantages of this model is that this model predicts really fast and is free of cost.

DISADVANTAGES:

- This model is may predict wrong predictions.
- CNN model take time to train in input is large.

8 APPLICATIONS

Following are some applications of Breat Cancer Risk Prediction model:

- Fast Prediction
- Free of cost
- highly applicable
- easy to use
- can be used in hospitals, homes

9 CONCLUSION

This Breast Cancer Risk Prediction System mainly focused in the advancement of predictive model to achieve good accuracy in predicting Breast Cancer outcomes using deep learning techniques like CNN. The Breast Cancer Risk Prediction System will be useful for the user who wishes to predict if they has Breast Cancer or not. This system classifies Breast Cancer into two cotegories benign or malignant. Users have feasibility to upload scanned images on a web page to know about the status.

10 FUTURE SCOPE

Future Scope of this Breast Cancer Risk Prediction System can be by adding the following to make it more advance: -

- Further research in this field should be carried out for the better performance of the classification techniques so that it can predict on more variables.
- New feature can be added like suggesting treatment. For example is the model predicts that user has Breast Cancer than treatment suggestions can also be veiwed on the web page.

11 BIBILOGRAPHY

1. Sivapriya J, Aravind Kumar V, Siddarth Sai S, Sriram S, Breast Cancer Prediction using Machine Learning, November 2019

2. Youness K, Mohammed B, Applying Best Machine Learning Algorithms for Breast Cancer Prediction and Classification , 2018

12 APPENDIX

A. Source code:

Breast_cancer_cnn.ipynb-

```
#importing the libraries
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
#initialize the model
model = Sequential()
model.add(Convolution2D(32,(3,3),input_shape = (64,64,3),activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
model.add(Flatten())
model.add(Dense(output_dim = 128 ,init = 'uniform',activation = 'relu'))
model.add(Dense(output_dim = 1,activation = 'sigmoid',init ='uniform'))
from keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range =
0.2,horizontal_flip = True)
test_datagen =ImageDataGenerator(rescale = 1./255)
x_train =
train_datagen.flow_from_directory('git/IISPS-INT-3342-Deep-Learning-Techniques-for-Breast-Cancer-
Risk-Prediction-using-Python/Dataset/trainset',target_size = (64,64),batch_size = 32, class_mode =
'binary')
x_test =
test_datagen.flow_from_directory('git/IISPS-INT-3342-Deep-Learning-Techniques-for-Breast-Cancer-R
isk-Prediction-using-Python/Dataset/testset',target_size = (64,64),batch_size = 32, class_mode =
'binary')
print(x_train.class_indices)
model.compile(loss = 'binary_crossentropy',optimizer = "adam",metrics = ["accuracy"])
model.fit_generator(x_train, steps_per_epoch = 32,epochs = 10,validation_data =
x_test,validation_steps = 8)
model.save("Breast_cancer.h5")
```

Breast_cancer_prediction.ipynb-

```
from keras.models  import load_model
from keras.preprocessing import image
model = load_model("Breast_cancer.h5")
```

```
img =
image.load_img("git/IISPS-INT-3342-Deep-Learning-Techniques-for-Breast-Cancer-Risk-Prediction-using-Python/Dataset/trainset/benign/8863_idx5_x1801_y2051_class0.png",target_size = (64,64))
import numpy as np
x  = image.img_to_array(img)
x = np.expand_dims(x,axis = 0)
pred = model.predict_classes(x)
pred
type(pred)
i = pred.flatten()
i
```

app.py-

```
from __future__ import division, print_function
# coding=utf-8
import sys
import os
import glob
import numpy as np
from keras.preprocessing import image
import swish_package
from swish_package import swish
from keras.applications.imagenet_utils import preprocess_input, decode_predictions
from keras.models import load_model
from keras import backend
from tensorflow.python.keras import backend
import tensorflow as tf
global graph
graph=tf.get_default_graph()
#global graph
#graph = tf.get_default_graph()
from skimage.transform import resize
# Flask utils
from flask import Flask, redirect, url_for, request, render_template
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
# Define a flask app
app = Flask(__name__)
# Model saved with Keras model.save()
```

```
MODEL_PATH = 'models/Breast_cancer.h5'
# Load your trained model
model = load_model(MODEL_PATH)
      # Necessary
# print('Model loaded. Start serving...')
# You can also use pretrained model from Keras
# Check https://keras.io/applications/
#from keras.applications.resnet50 import ResNet50
#model = ResNet50(weights='imagenet')
#model.save('')
print('Model loaded. Check http://127.0.0.1:5000/')
@app.route('/', methods=['GET'])
def index():
   # Main page
   return render_template('index.html')

@app.route('/predict', methods=['GET', 'POST'])
def upload():
   if request.method == 'POST':
      # Get the file from post request
      f = request.files['file']

      # Save the file to ./uploads
      basepath = os.path.dirname(__file__)
      file_path = os.path.join(
         basepath, 'uploads', secure_filename(f.filename))
      f.save(file_path)
      img = image.load_img(file_path, target_size=(64, 64))
      x = image.img_to_array(img)
      x = np.expand_dims(x, axis=0)

      with graph.as_default():
         preds = model.predict_classes(x)
         i = preds.flatten()
      index = ['benign','malignant']
      text = "prediction : "+ index[i[0]]
           # ImageNet Decode
```

```
    return text
if __name__ == '__main__':
app.run(debug=False,threaded = False)
```

base.html-

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible"content="ie=edge">
  <title>Artificial Intelligence</title>
  <link
href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"rel="stylesheet">
  <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
  <link href="{{ url_for('static', filename='css/main.css') }}"rel="stylesheet">
</head>
<body>
  <nav class="navbar navbar-dark bg-dark">
    <div class="container">
      <a class="navbar-brand"href="#">Artificial Intelligence</a>
      <button class="btn btn-outline-secondary my-2
my-sm-0"type="submit">Help</button>
    </div>
  </nav>
  <div class="container">
    <div id="content"style="margin-top:2em">{% block content %}{% endblock %}</div>
  </div>
</body>
<footer>
  <script src="{{ url_for('static', filename='js/main.js') }}"type="text/javascript"></script>
</footer>
</html>
```

index.html-

```
{% extends "base.html" %} {% block content %}

<h2>Breast Cancer Risk Prediction</h2>
<div>
  <form id="upload-file"method="post"enctype="multipart/form-data">
    <label for="imageUpload"class="upload-label">
      Choose...
    </label>
    <input type="file"name="file"id="imageUpload"accept=".png, .jpg, .jpeg">
  </form>
  <div class="image-section"style="display:none;">
    <div class="img-preview">
      <div id="imagePreview">
      </div>
    </div>
    <div>
      <button type="button"class="btn btn-primary btn-lg
"id="btn-predict">Predict!</button>
    </div>
  </div>
  <div class="loader"style="display:none;"></div>
  <h3 id="result">
    <span></span>
  </h3>
</div>

{% endblock %}
```

B. UI output Screenshot:

Below are a few screen shots of the working Breast Cancer Risk Prediction Model:

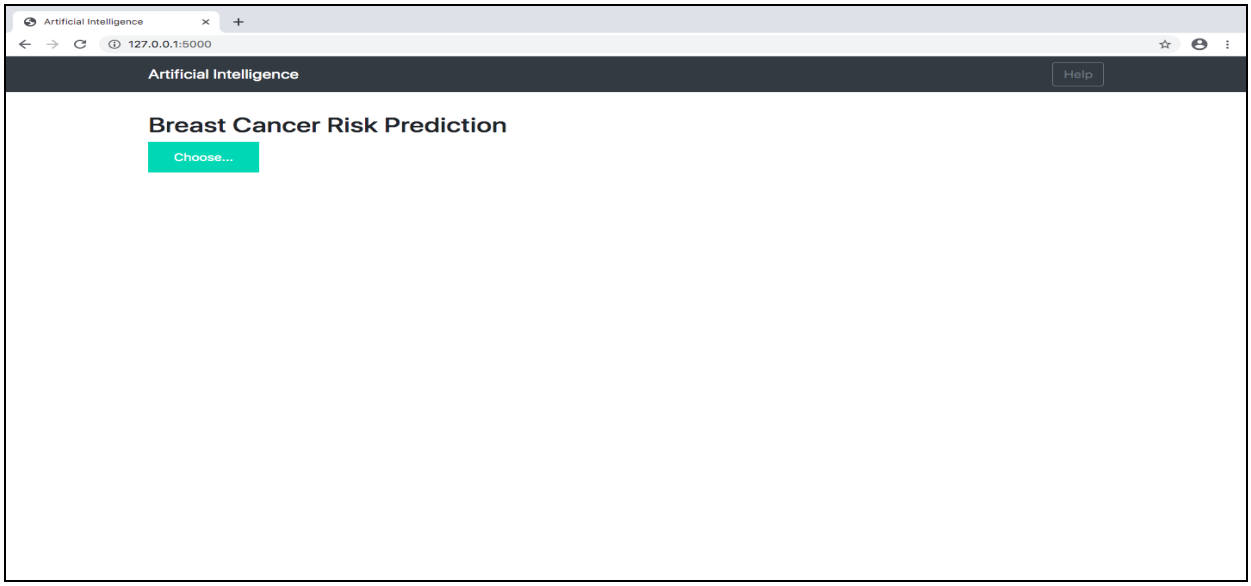


fig. 3(a)

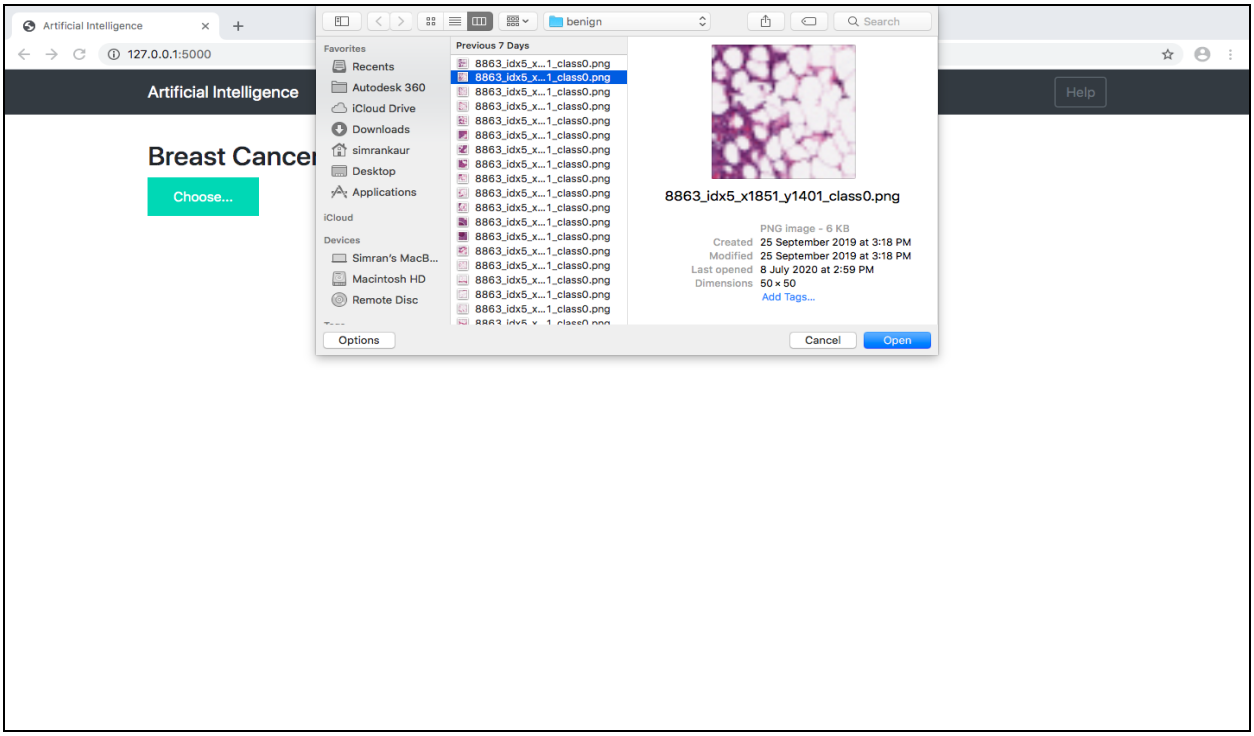


fig. 3(b)

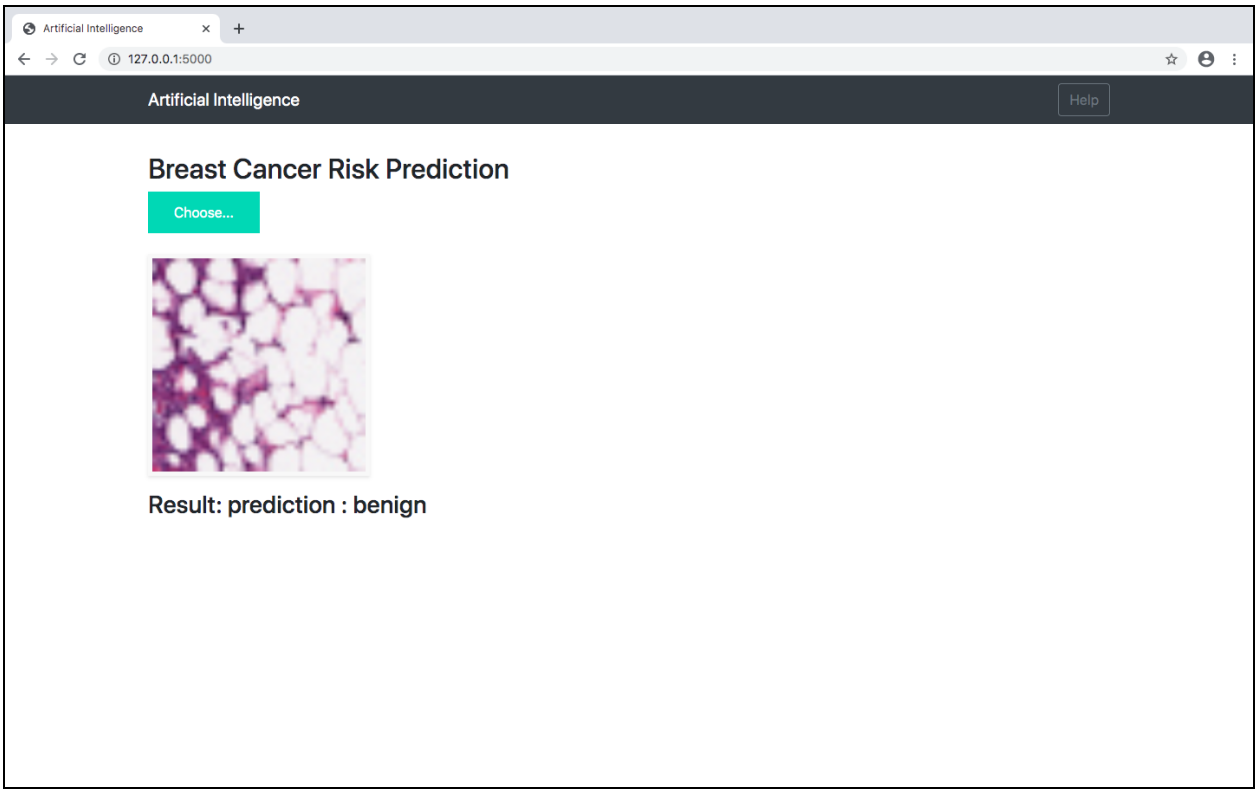


fig. 3(c)

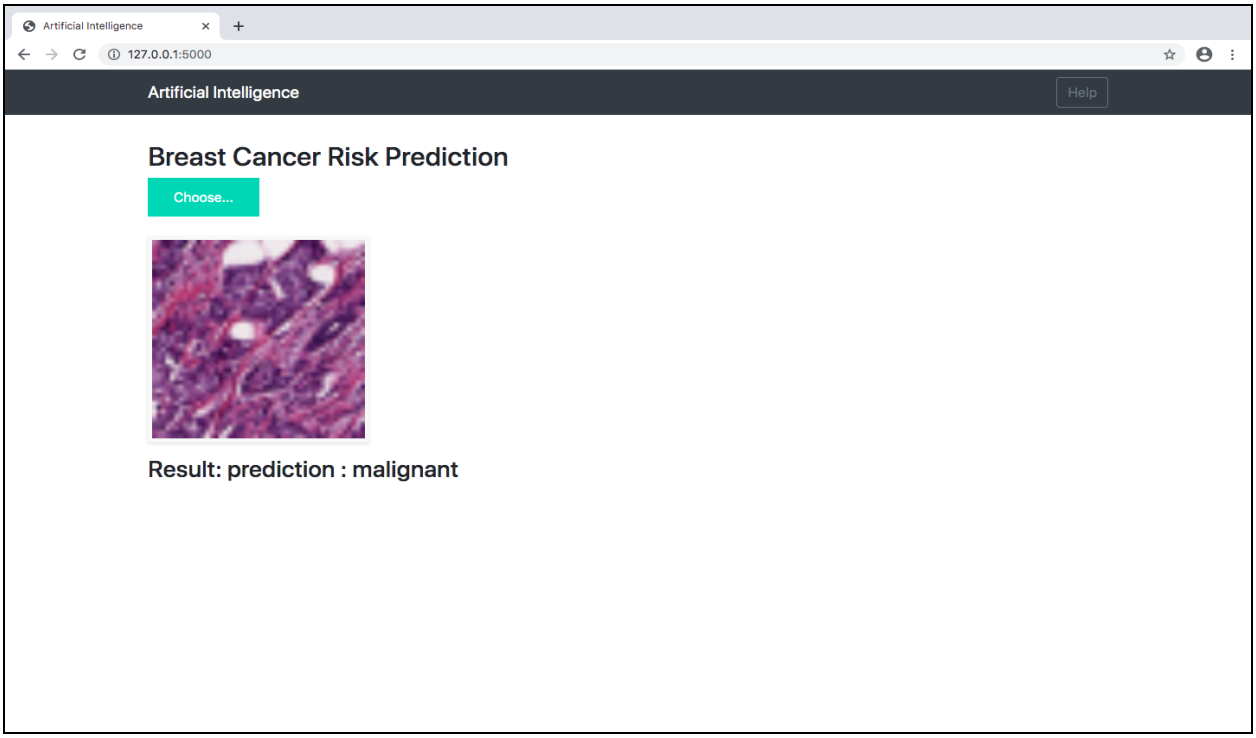


fig. 3(d)

Fig. 3(a), 3(b), 3(c) and 3(d) display the working User Interface of the Breat Cancer Risk Prediction Model