SMARTBRIDGE
Let's Bridge the Gap

Project Report

Project Title: **Deep Learning Techniques For Breast Cancer Risk Prediction Using Python**

Work Submitted By : M SAI PRACHETHANA

MENTORS:
Mr. Rammohan Bethi
Mr. Vinay kumar
Mr.Kalkeseetharaman PK

PROJECT DETAILS:

Internship Title	:	RSIP Career Basic AI 056
Project ID	:	SPS_PRO_202
Project Title	:	Deep Learning Techniques for Breast Cancer Risk Prediction using Python
Duration	:	4 Weeks
Kickoff Date:	:	June 8th,2020
Finish Date:	:	July 6th,2020

ACKNOLEDGEMENMT

This project has taken a considerable amount of time and resources. I would like to acknowledge the help of all of those who have made this project possible. In finical I would like to thank my supervisor Mr. Rammohan Bethi Gahlot for his time, patience and to these people I would like to thank the members of the Smart bridge career workshop for their technical help in setting up various codes and faults. Also, I would like to thank all my co-interns who have worked on the open source projects without whose on the open source projects without whose efforts this project would not have been possible

1. Introduction

1.1 Overview

In this project 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 convolution 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.

1.2. Purpose

Breast cancer is one of the main causes of cancer death worldwide. Early diagnostics significantly increases the chances of correct treatment and survival, but the process tedious and often leads to a disagreement between pathologists. Computer-aided diagnosis system showed potential for improving diagnostic accuracy. But early detection and prevention can significantly reduce the chances of death. It is important to detect breast cancer as early as possible

2. Literature Survey

2.1. Existing Problem

Breast cancer occur when cells in the breast become abnormal and divide uncontrollably. Breast cancer usually starts in the glands that produce milk or the tubes that carry a milk from the glands to the nipple

Invasive breast cancer will be diagnosed in about 268,600 women

Noninvasive will be diagnosed in almost 62,930.Almost 41,760 women were died

Breast disorder may be noncancerous or cancerous. Most are noncancerous and not life threatening. Often, they do not require treatment. In contrast, breast cancer can mean loss of a breast or of life. Thus, for many women, breast cancer is their worst fear. However, potential problems can often be detected early when women regularly by their doctor, and have mammogram as recommended. Early detection of breast cancer can be essential to successful treatment

2.2. Proposed problem

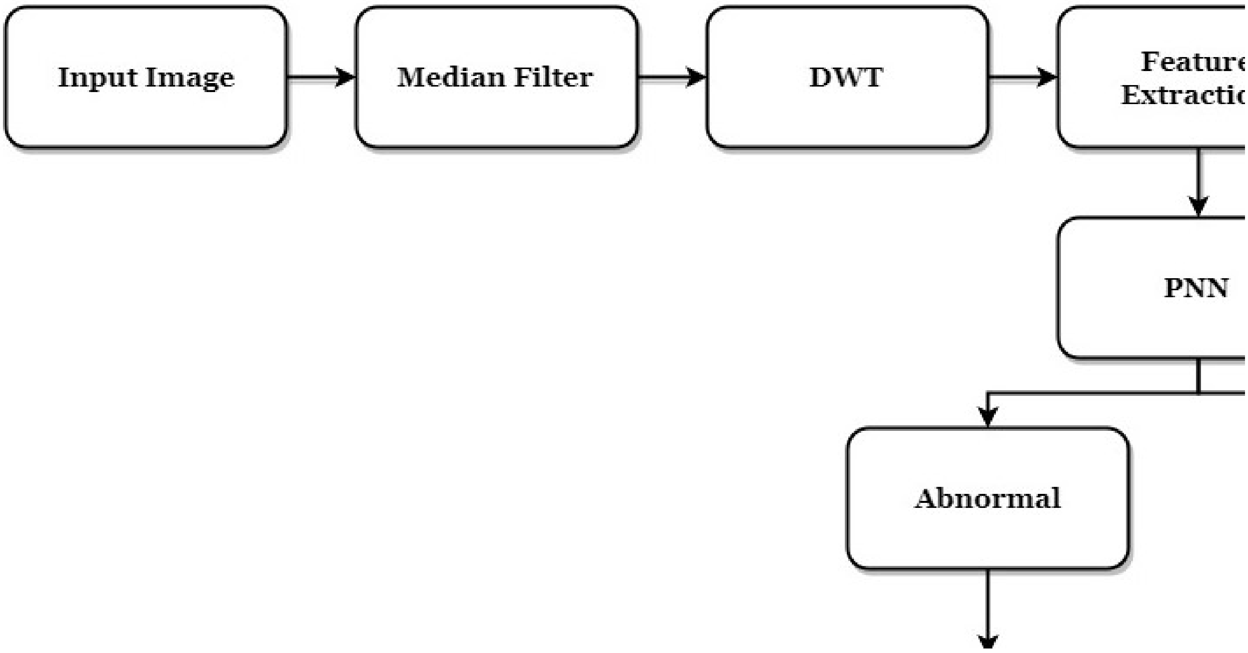
The accurate prediction of survival rate in patients with breast cancer remains challenge due to the increasing complexity of cancer, treatment protocols, and various patient population samples. Reliable and well-validated predictions could assist in a better way personalized care and treatment, and improve the control over the cancer development. Usually in good clinic practices, clinicians use data collected from different sources as medical records, clinical laboratory tests, and studies aiming more precise diagnostics, therapy and disease-development prognosis

Prediction and prognosis of cancer development are focused on three major domains: risk assessment or prediction of cancer susceptibility, prediction of cancer relapse, and prediction of cancer survival rate. The first domain comprises prediction of the probability of developing certain cancer prior to the patient diagnostics. The second issue is related to prediction of cancer recurrence in terms of diagnostics and treatment, and the third case is aimed at prediction of cancer possible parameters characterizing cancer development and treatment after the diagnosis of the disease:

survival time, life expectancy, progression, drug sensitivity, etc. The survivability rate and the cancer relapse are dependent very much on the medical treatment and the quality of the diagnosis.

3. Theoretical Analysis

3.1. Block diagram



3.2. Hardware/Software Designing

Hardware:

Manufacture	: Microsoft Corporation
Processor	: Intel@Corei7-8650u cpu@1.90GHz 2.11GHz
Installed memory(RAM)	: 16.0GB System Type:64-bit Operating System,x64-based Processor

Software:

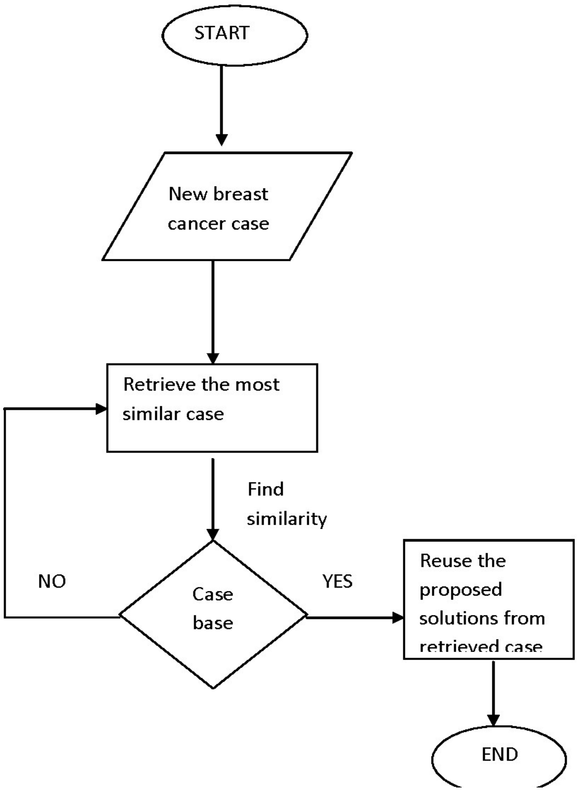
Operating System	: Window 10pro
Anaconda version	: 3.7 for windows
Python version	: 3.6
Tensor flow version	: 1.14.0 using anaconda
Keras version	: 2.2.4 using anaconda

4. Experimental Investigations

The classification methods are used to classify suspicious areas of breast DCE-MR images into benign or malignant tissue. The selection of the correct classifier is a factor that very sensitively affects the performance of the correct classification. The selected classifier needs to fit well to the training data (i.e. produce good separation between the classes) while, on the other hand, being as robust as possible to unseen data (generalizability). In this section, Three types of classifiers such as artificial neural networks, support vector machines and bee colony optimization algorithm trained neural network classifier were evaluated for classification of the suspicious lesions in breast MRI and to find an optimal classifier that outperforms all other explored classifiers for the examined dataset of breast DCE –MR images.

ANN is a powerful classifier that represents a complex input/ output relationships. It resembles the human brain in acquiring knowledge through learning and storing knowledge within inter-neuron connection strengths.

5. Flowchart


START
New breast cancer case
Retrieve the most similar case
Find similarity
NO
Case base
YES
Reuse the proposed solutions from retrieved case
END

6. Result

A new lump or mass

The most common sign of breast cancer is a new lump or mass in breast tissue. This can be painful or painless. It may be irregularly shaped.

Skin irritation

A long with swelling and redness, your skin can look irritated. Sometimes it can look dimpled, like an orange peel.

Thickening skin

Certain types of breast cancer can cause significant thickness of the skin along your breasts.

Underarm swelling and pain

Something the original site of breast cancer can spread to lymph nodes near the underarm area, causing pain and swelling.

Weakness and fatigue

As breast cancer progresses you may feel extremely weak and fatigued.

Unintentional weight loss

Breast cancer can decrease you appetite leading to unintentional weight loss

Nipple retraction

Breast cancer can cause the nipple to turn inward and away from the outer breasts

Change in breast color

Your breasts can change color during the presence of cancer

Breast pain

The pain from a new lump can expand to your entire breast

7. Advantages and disadvantages

Advantages

Breast cancer screening every 2 years over a 20-year period:

1. Reduces the risk of dying from breast cancer

Of 1,000 women who have a mammogram every 2 years for 20 years, 7 deaths are prevented

2. Reduces the risk of having to undergo chemotherapy

Screening often allows for the detection of cancers at an early stage of development. Treatment is then possible without chemotherapy.

3. Allows women to know the health of their breasts

The vast majority of women (nearly 98%) will not have breast cancer if their mammograms and additional examinations do not reveal cancers.

Disadvantages

Breast cancer screening every 2 years for 20 years can lead to:

1. Periods of waiting and anxiety when additional examinations are required

Almost half the women who participate in the screening for 20 years (453 in 1,000) have at least one additional examination.

This represents 156 more women than in the 1,000 who do not participate in the screening.

2. Possible over diagnosis

Of 77 breast cancer diagnoses, 10 would be cases of over diagnosis

8. Applications

The Breast Cancer e-Support program (BCS) is a mobile IIP application (App) for Chinese women with breast cancer who are undergoing chemotherapy. The BCS was developed using the theoretical framework [20], which incorporate Bandera's self-efficacy theory [21] and the social exchange theory [22].

1. Cancer coach

Help manage your personal cancer journey with information specifies to your diagnosis. Includes calendar, note-taking, audio-recording and question to ask your doctor

2. Pills on the go

A pill reminder for people on the go. Get an alarm each time it's medication time

3. Breast check reminder

Self-exams save lives. If you have a difficult time remembering your monthly breast exam, let this app remainder

4. Caring Bridge

Caring Bridge connects you with the support of friends and family when it's needed most

9. Conclusion

A plan for the diagnosis and treatment of cancer is a key component of any overall cancer control plan. Its main goal is to cure cancer patients or prolong their life considerably, ensuring a good quality of life. In order for a diagnosis and treatment programmer to be effective, it must never be developed in isolation. It needs to be linked to an early detection programmer so that cases are detected at an early stage, when treatment is more effective and there is a greater chance of cure. It also needs to be integrated with a palliative care programmer, so that patients with advanced cancers, who can no longer benefit from treatment, will get adequate relief from their physical, psychosocial and spiritual suffering. Furthermore, programmers should include a awareness-raising component, to educate patients, family and community members about the cancer risk factors and the need for taking preventive measures to avoid developing cancer.

Where resources are limited, diagnosis and treatment services should initially target all patients presenting with curable cancers, such as breast, cervical and oral cancers that can be detected early. They could also include childhood acute lymphatic leukemia, which has a high potential for cure although it cannot be detected early. Above all, services need to be provided in an equitable and sustainable manner. As and when more resources become available, the programmer can be extended to include other curable cancers as well as cancers for which treatment can prolong survival considerably.

10. Future Scope

Breast cancer is the most common malignancy in women with post-operative recurrence and metastases acting as the leading cause of breast-cancer associated mortality. The number of patients in post-treatment surveillance programs is increasing secondary to the survival benefit of screening mammography and adjuvant therapies. After curative primary treatment, approximately 15% of breast cancer survivors will develop a second breast malignancy within ten years. This risk is further compounded by personal characteristics such as age and family history.

Follow-up care after primary breast cancer treatment includes physical and psychological rehabilitation, assessment of treatment efficacy, and detection of recurrent or metachronous cancers. Current national comprehensive cancer Network (NCCN) guidelines recommend a history and physical examination every 4-6 months for 5 years, then every 12 months.

SOURCE CODE:

Breast Cancer Prediction.ipynb

In [1]:

```
from keras.preprocessing.image import ImageDataGenerator
```

```
Using TensorFlow backend.
```

In [2]:

```
train_datagen = ImageDataGenerator(rescale =1./255, shear_range = 0.2,
zoom_range=0.2, horizontal_flip =True)test_datagen =
ImageDataGenerator(rescale=1./255)
```

In [3]:

```
x_train=train_datagen.flow_from_directory(r'C:\Users\DELL\Desktop\Intership csv
files\Project\Train',target_size=(64,64),batch_size=32,class_mode='categorical')x_te
st=test_datagen.flow_from_directory(r'C:\Users\DELL\Desktop\Intership csv
files\Project\Test',target_size=(64,64),batch_size=32,class_mode='categorical')print(
x_train.class_indices)
```

```
Found 400 images belonging to 2 classes.

Found 120 images belonging to 2 classes.

{'CANCER': 0, 'NORMAL': 1}
```

In [4]:

```
import tensorflow as tftf.compat.v1.get_default_graph
```

Out[4]:

```
<function tensorflow.python.framework.ops.get_default_graph()>
```

In [6]:

```
from keras.models import Sequentialfrom keras.layers import Densefrom
keras.layers import Convolution2Dfrom keras.layers import MaxPooling2Dfrom
keras.layers import Flatten
```

In [7]:

```
model = Sequential()
```

In [8]:

```
model.add(Convolution2D(32,(3,3),input_shape = (64,64,3),activation = 'relu'))
```

In [9]:

```
model.add(MaxPooling2D(pool_size = (2,2)))model.add(Flatten())
```

In [10]:

```
model.add(Dense(output_dim = 128 ,init = 'uniform',activation =
'relu'))model.add(Dense(output_dim = 2,activation = 'softmax',init ='uniform'))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: UserWarning:
Update your `Dense` call to the Keras 2 API: `Dense(activation="relu", units=128,
kernel_initializer="uniform")`

  """Entry point for launching an IPython kernel.

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: UserWarning:
Update your `Dense` call to the Keras 2 API: `Dense(activation="softmax", units=2,
kernel_initializer="uniform")`
```

In [13]:

```
model.compile(loss = 'categorical_crossentropy',optimizer = "adam",metrics =
["accuracy"])
```

In [15]:

```
model.fit_generator(x_train, steps_per_epoch = 400,epochs =10 ,validation_data =
x_test,validation_steps = 120)
```

Out[15]:

```
Epoch 1/10

400/400 [==============================] - 91s 227ms/step - loss: 0.5100 -
accuracy: 0.7746 - val_loss: 0.8759 - val_accuracy: 0.4333
```

```
Epoch 2/10

400/400 [==============================] - 88s 220ms/step - loss: 0.1338 -
accuracy: 0.9597 - val_loss: 1.0125 - val_accuracy: 0.4667

Epoch 3/10

400/400 [==============================] - 90s 226ms/step - loss: 0.0812 -
accuracy: 0.9772 - val_loss: 1.4543 - val_accuracy: 0.4583

Epoch 4/10

400/400 [==============================] - 88s 221ms/step - loss: 0.0549 -
accuracy: 0.9864 - val_loss: 1.9000 - val_accuracy: 0.4500

Epoch 5/10

400/400 [==============================] - 86s 215ms/step - loss: 0.0388 -
accuracy: 0.9915 - val_loss: 1.5310 - val_accuracy: 0.4583

Epoch 6/10

400/400 [==============================] - 87s 219ms/step - loss: 0.0259 -
accuracy: 0.9959 - val_loss: 2.3786 - val_accuracy: 0.4917

Epoch 7/10

400/400 [==============================] - 87s 218ms/step - loss: 0.0207 -
accuracy: 0.9959 - val_loss: 2.7610 - val_accuracy: 0.4833

Epoch 8/10

400/400 [==============================] - 87s 218ms/step - loss: 0.0134 -
accuracy: 0.9989 - val_loss: 2.3245 - val_accuracy: 0.4917

Epoch 9/10

400/400 [==============================] - 87s 218ms/step - loss: 0.0115 -
accuracy: 0.9978 - val_loss: 2.2138 - val_accuracy: 0.5000

Epoch 10/10

400/400 [==============================] - 89s 221ms/step - loss: 0.0075 -
accuracy: 0.9999 - val_loss: 2.1935 - val_accuracy: 0.4833

Out[15]:

<keras.callbacks.callbacks.History at 0x18c6a6103c8>

In [23]: model.save("cancer.h5")
```

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

from keras.applications.imagenet_utils import preprocess_input, decode_predictions

from keras.models import load_model

from keras import backend

from tensorflow.keras import backend

import tensorflow as tf

global graph

graph=tf.compat.v1.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/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('base.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(x)

        index = ['CANCER','NORMAL']
        text = "prediction : "+index[preds[0]]

          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>Breast Cancer</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="#">Breast Cancer Detection</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"><h2>Please Upload an
Image</h2>

<div>
    <form id="upload-file" method="post" enctype="multipart/form-data">
        <label for="imageUpload" class="upload-label">
            Upload
        </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></div>
        </div>
</body>


<footer>
        <script src="{{ url_for('static', filename='js/main.js') }}"
type="text/javascript"></script>
</footer>
</html>
```