Deep Learning Techniques for Breast Cancer Risk Prediction

1.INTRODUCTION:

1.1 Overview:

Breast cancer is one of the main causes of cancer death worldwide. Computer-aided diagnosis systems showed the 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.

The goal is to classify images into two classifications of malignant and benign. As early diagnostics significantly increases the chances of correct treatment and survival. In this application, we are helping the doctors and patients to classify the Type of Tumour for the specific image given with the help of Neural Networks.

Here we present deep-learning techniques for healthcare, centering our discussion on deep learning in computer vision, natural language processing, reinforcement learning, and generalized methods. We describe how these computational techniques can impact a few key areas of medicine and explore how to build end-to-end systems. Our discussion of computer vision focuses largely on medical imaging, and we describe the application of natural language processing to domains such as electronic health record data. Similarly, reinforcement learning is discussed in the context of robotic-assisted surgery, and generalized deep-learning methods for genomics are reviewed.

1.2 Purpose:

Breast cancer is the second leading cause of cancer deaths among U.S. women1 and screening mammography has been found to reduce mortality2.Despite the benefits, screening mammography is associated with a high risk of false positives as well as false negatives. The average sensitivity of digital screening mammography in the U.S. is 86.9% and the average specificity is 88.9%3. To improve the predictive accuracy of screening mammography To evaluate the accuracy and efficiency of early breast cancer detection, Hence, the requirement of time is to develop the technique which gives minimum error to increase accuracy.

2.LITERATURE SURVEY:

2.1 Existing Problem:

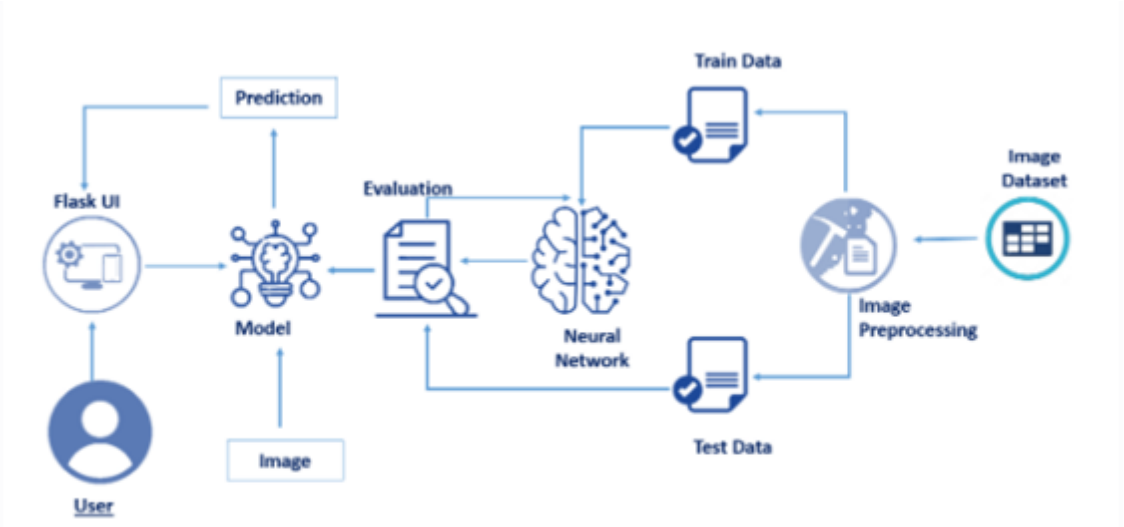
Breast cancer is one of the deadliest disease, is the most common of all cancers and is the leading cause of cancer deaths in women worldwide, accounting for 1.6% of deaths and case fatality rates are highest in low-resource countries Women are seriously threatened by breast cancer with high morbidity and ortality. The lack of robust prognosis models results in difficulty for doctors to prepare a treatment plan that may prolong patient survival time.

2.2 Proposed Solution:

Developing a model using deep learning algorithm trained on a linked dataset of mammograms and electronic health records and also integrating it into a web application so that normal users can also use it. This helps in evaluate the accuracy and efficiency of a deep learning approach for early breast cancer detection, accuracy comparable to radiologists

3.THEORITICAL ANALYSIS:

3.1Block Diagram:



3.2Hardware/Software designing:

Requirments:

- Operating system of our choice
- A desktop or laptop
- Prefered text editor
- Programming language i.e Python3 or upward installed in our system.

Implementations:

1. install NumPy and rename it as np
   pip install NumPy as np

2. install pandas and rename it as pd
   pip install pandas as pd

3. install matplotlib.pyplot and rename it as matplotlib.plt
   pip install matplotlib.pyplot as plt
4. download the breast cancer dataset
5. read the CSV file and put it in a variable
   example – data = read.csv("../input/data.csv")

6. drop the columns which are unnamed or which are not needed in prediction or doesn't have any values associated with it
7. denote malignant as M and benign as B
8. convert the M and B to an integer value
9. if the value is greater than is between the range 1-5 denote it as malignant and the value is above 6, denote it as benign.
10. if diagnosed with malignant cancer denote it with 1 else 0
11. plot the graph using lmplot and watch the distribution of malignant cancer(1) and benign cancer(0)
12. take 2 variables. One is for input and other for output. Let x be input and y be output.

13. Split the data into test data and training data
    Use the code from sklearn.model_selection import train_test_split X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.33, random_state = 42)

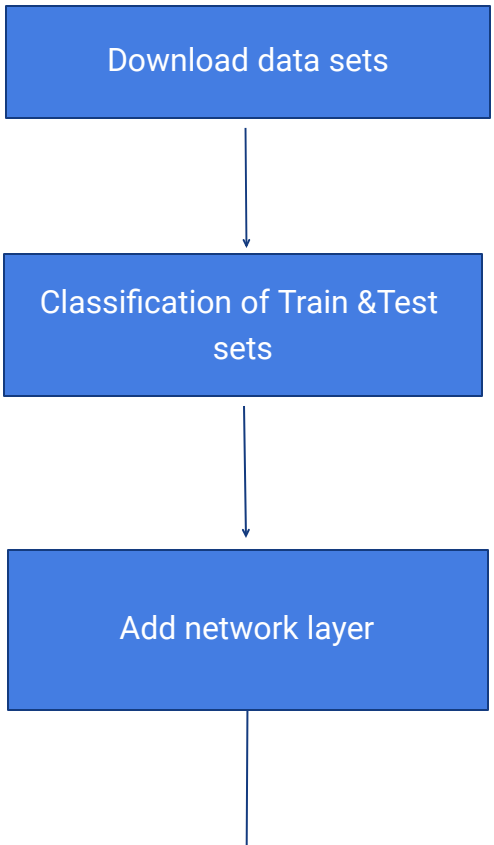
14. Check the prediction score on test data using KNN
    Knn.score(X_test,y_test)

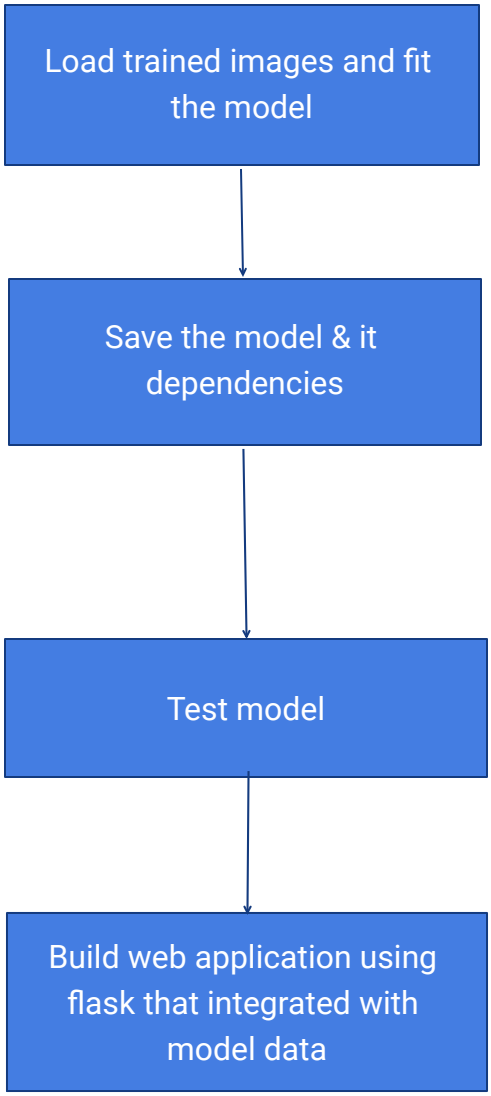
15. Perform cross-validation
16. Create a confusion matrix to check the correctness of the prediction.

4.EXPERIMENTAL INVESTIGATIONS:

1. Martin AM, Weber BL: Genetic and hormonal risk factors in breast cancer. J Natl Cancer Inst 92:1126-1135, 2000
2. Rebbeck TR, Lynch HT, Neuhausen SL, et al: Prophylactic oophorectomy in carriers of BRCA1 or BRCA2 mutations. N Engl J Med 346:1616- 1622, 2002
3. Kauff ND, Satagopan JM, Robson ME, et al: Risk-reducing salpingooophorectomy in women with a BRCA1 or BRCA2 mutation. N Engl J Med 346:1609-1615, 2002

5.FLOW CHART:





6.RESULT:

By applying the CNN algorithm on the preprocessed images of data sets and ,With the help of deep neural network structure we can able to predict the cancer is benign or malignan. Finally the build web applications using the Flask framework helps the normal users, so that they can also use it. The users need to give the X-ray images to know the predictions.

7.ADVANTAGES & DISADVANTAGES:

Advantages:

- we can increase the accuracy of our prediction.
- Efficient at delivering High-quality Results

Disadvantages:

- The Need for Lots of Data
- Neural Networks at the Core of Deep Learning are Black Boxes
- Lack of Flexibility
- Overfitting the Model

8.APPLICATIONS:

Breast cancer risk assessment provides an estimation of disease risk that can be used to guide management for women at all levels of risk. In addition, the likelihood that breast cancer risk is due to specific genetic susceptibility (such as BRCA1 or BRCA2 mutations) can be determined. Recent developments have reinforced the clinical importance of breast cancer risk assessment.
Tamoxifen chemoprevention as well as prevention studies such as the Study of Tamoxifen and Raloxifene are available to women at increased risk of developing breast cancer. In addition, specific management strategies are now defined for BRCA1 and BRCA2 mutation carriers.
Risk may be assessed as the likelihood of developing breast cancer (using risk assessment models) or as the likelihood of detecting a BRCA1 or BRCA2 mutation (using prior probability models). Each of the models has advantages and disadvantages, and all need to be interpreted in context.
We review available risk assessment tools and discuss their application. As illustrated by clinical examples, optimal counseling may require the use of several models, clinical judgment, to provide the most accurate and useful information to women and their families.

9.CONCLUSION:

There was a striking improvement in the accuracy of classification of women with and without breast cancer achieved with Deep natural network compared to the state-of-the-art model-based approaches.
It is remarkable to see the success of deep learning in such varied real world problems. In this blog, I have demonstrated how to classify benign and malignant breast cancer from a collection of microscopic images using convolutional neural networks and transfer learning.

10.FUTURE SCOPE:

AI is set to change the medical industry in the coming decades it wouldn't make sense for pathology to not be disrupted too.Currently, Deep Learning models are still in the testing and experimentation phase for cancer prognoses.
As datasets are getting larger and of higher quality, researchers are building increasingly accurate models.

Here's what a future cancer biopsy might look like:

You perform clinical tests, either at a clinic or at home. Data is inputted into a pathological ML or Deep Learning system. A few minutes later, you receive an email with a detailed report that has an accurate prediction about the development of your cancer. While you might not see AI doing the job of a pathologist today, you can expect Deep Learning or MI to replace your local pathologist in the coming decades, and it's pretty exciting!
Deep Learning models still have a long way to go, most models still lack sufficient data and suffer from bias. Yet, something we are certain of is that Deep Learning is the next step of pathology, and it will disrupt
the industry.

11.BIBILOGRAPHY:

Steyerberg, Ewout W. *Clinical Prediction Models*. Springer Science & Business Media, 16 Dec. 2008.

Wyld, Lynda, et al. *Breast Cancer Management for Surgeons*. Springer, 2017.

Council, National Research, et al. *Saving Women's Lives*. National Academies Press, 2005.

From the University of Pennsylvania Cancer Center; Abramson Family Cancer Research Institute, Philadelphia, PA; and Hamilton Regional Cancer Centre, Hamilton, Ontario, Canada. Submitted July 1, 2002; accepted October 24, 2002.

APPENDIX:

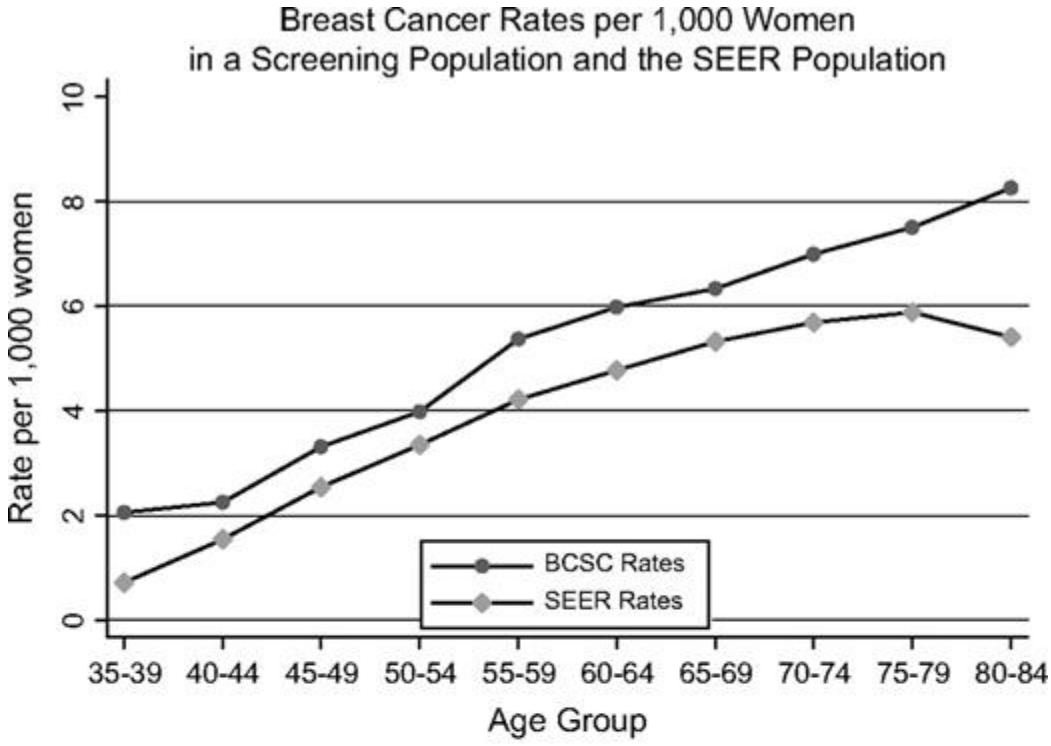
There were 2 884 197 screening mammograms, but we included only the 2 392 998 (83.0%) index screening mammograms from 1 007 600 women who had had a previous mammogram in the prior 5 years.

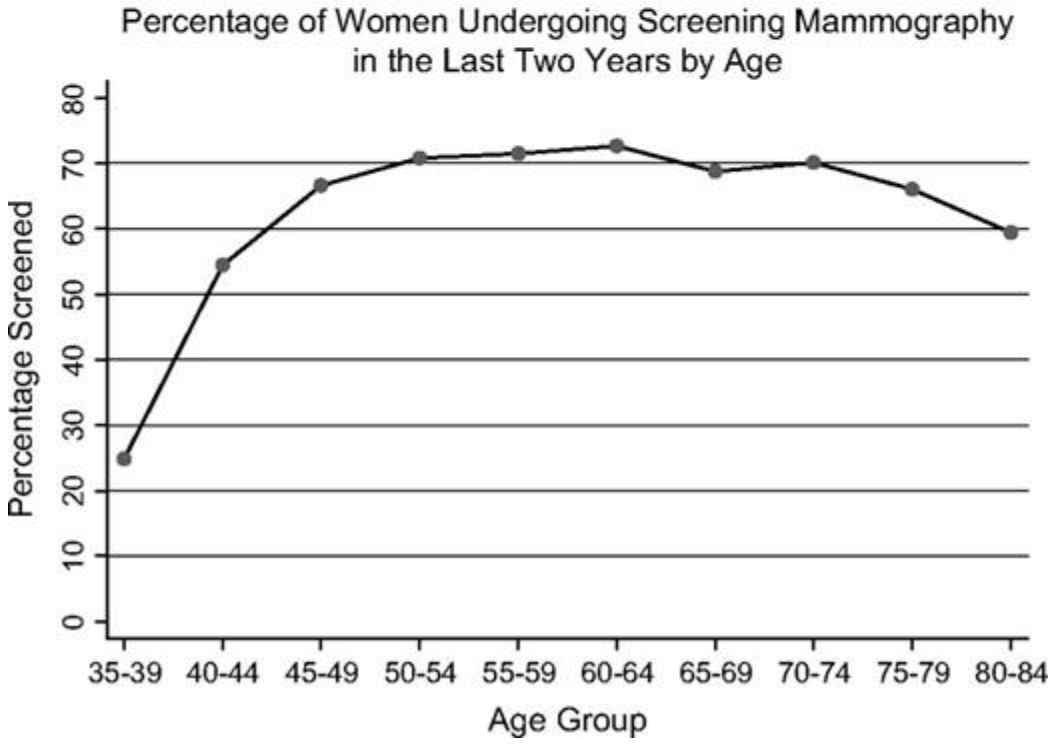
Breast cancer was diagnosed within 1 year of a screening mammogram in 11 638 women, for an absolute rate of 4.86 breast cancers per 1000 screening mammograms (95% CI = 4.78 to 4.95). Most (75.7%) of these 11 638 breast cancers were diagnosed within 3 months of the screening mammogram.

Risk factors vary by menopausal status, so that separate models were fit for premenopausal and postmenopausal women.

This procedure required excluding 7.6% of the mammograms from women aged 45–54 years with unknown menopausal status. The remaining mammograms were classified as premenopausal (n = 568 215; 25.7%) or postmenopausal (n = 1 642 824; 74.3%). There were 1726 breast cancers among premenopausal women, for an absolute rate of

3.04 per 1000 screening mammograms (95% CI = 2.89 to 3.18), and 9300 breast cancers among postmenopausal women, for an absolute rate of 5.66 per 1000 screening mammograms (95% CI = 5.55 to 5.78)


Breast Cancer Rates per 1,000 Women
in a Screening Population and the SEER Population
Rate per 1,000 women
0
2
4
6
8
10
BCSC Rates
SEER Rates
35-39
40-44
45-49
50-54
55-59
60-64
65-69
70-74
75-79
80-84
Age Group


Percentage of Women Undergoing Screening Mammography
in the Last Two Years by Age
Percentage Screened
0
10
20
30
40
50
60
70
80
35-39
40-44
45-49
50-54
55-59
60-64
65-69
70-74
75-79
80-84
Age Group

SOURCE CODE:

```
from keras.preprocessing.image import ImageDataGenerator
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
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1)

x_train=train_datagen.flow_from_directory('breastcancerdataset/train',target_size=(64,64),batch_size=32,class_mode='binary')
x_test=test_datagen.flow_from_directory('breastcancerdataset/test',target_size=(64,64),batch_size=32,class_mode='binary')
Found 103 images belonging to 2 classes.
Found 22 images belonging to 2 classes.

print(x_train.class_indices)
{'benign': 0, 'malignant': 1}

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(units=40,kernel_initializer='uniform',activation ='relu'))
model.add(Dense(units=1,activation='softmax',kernel_initializer='uniform'))
model.compile(loss='binary_crossentropy',optimizer="adam",metrics= ["accuracy"])
model.fit_generator (x_train,steps_per_epoch = 5,epochs = 10,validation_data = x_test,validation_steps = 40)
Epoch 1/10
5/5 [==============================] - 18s 4s/step - loss: 11.8570 - acc: 0.2563 - val_loss: 11.5945 - val_acc: 0.2727
Epoch 2/10
5/5 [==============================] - 17s 3s/step - loss: 11.7427 - acc: 0.2634 - val_loss: 11.5945 - val_acc: 0.2727
Epoch 3/10
```

```
5/5 [==============================] - 17s 3s/step - loss: 11.5604 - acc: 0.2749 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 4/10
5/5 [==============================] - 17s 3s/step - loss: 11.9443 - acc: 0.2508 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 5/10
5/5 [==============================] - 17s 3s/step - loss: 11.9481 - acc: 0.2505 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 6/10
5/5 [==============================] - 17s 3s/step - loss: 11.5488 - acc: 0.2756 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 7/10
5/5 [==============================] - 17s 3s/step - loss: 12.4501 - acc: 0.2191 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 8/10
5/5 [==============================] - 17s 3s/step - loss: 11.7549 - acc: 0.2627 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 9/10
5/5 [==============================] - 16s 3s/step - loss: 11.9481 - acc: 0.2505 - val_loss:
11.5945 - val_acc: 0.2727
Epoch 10/10
5/5 [==============================] - 17s 3s/step - loss: 11.4577 - acc: 0.2813 - val_loss:
11.5945 - val_acc: 0.2727
```

Out[49]:

```
<keras.callbacks.History at 0x16fc118a2e8>
```

```
model.save("breastcancer.h5")
from keras.models import load_model
from keras.preprocessing import image
import numpy as np
model=load_model("breastcancer.h5")
img=image.load_img('benign.png',target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred=model.predict_classes(x)
pred
```

```
array([[1]])
```

```
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.get_default_graph()
from skimage.transform import resize
from flask import Flask, redirect, url_for, request, render_template
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
app = Flask(__name__)
model = load_model("breastcancer.h5")
print('Model loaded. Check http://127.0.0.1:5000/')
```
Model loaded. Check http://127.0.0.1:5000/

```
@app.route('/', methods=['GET'])
def index():
    # Main page
    return render_template('bcancer.html')
@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['image']

        # 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)
        if preds[0][0]==0:
            text = "The tumor is benign.. Need not worry!"
        else:
            text = "It is a malignant tumor... Please Consult Doctor"
        text = text
                # ImageNet Decode
         return text
if __name__ == '__main__':
    app.run(debug=True,threaded = False)
```

```
 Debugger is active!
 Debugger PIN: 130-434-379
 Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```