

Deep Learning Techniques For Breast Cancer Risk Prediction Using IBM Cloud

Developer Name: Logeswaran K

Category: Deep Learning

Skills Required: CNN, Flask Integration, Numpy, Tensorflow, Keras, IBM Watson Studio, Open CV, Deep Learning

Implementation Coding:

Deployment coding can be found at below GitHub Repo link

<https://github.com/smartinternz02/SI-GuidedProject-6117-1635490836>

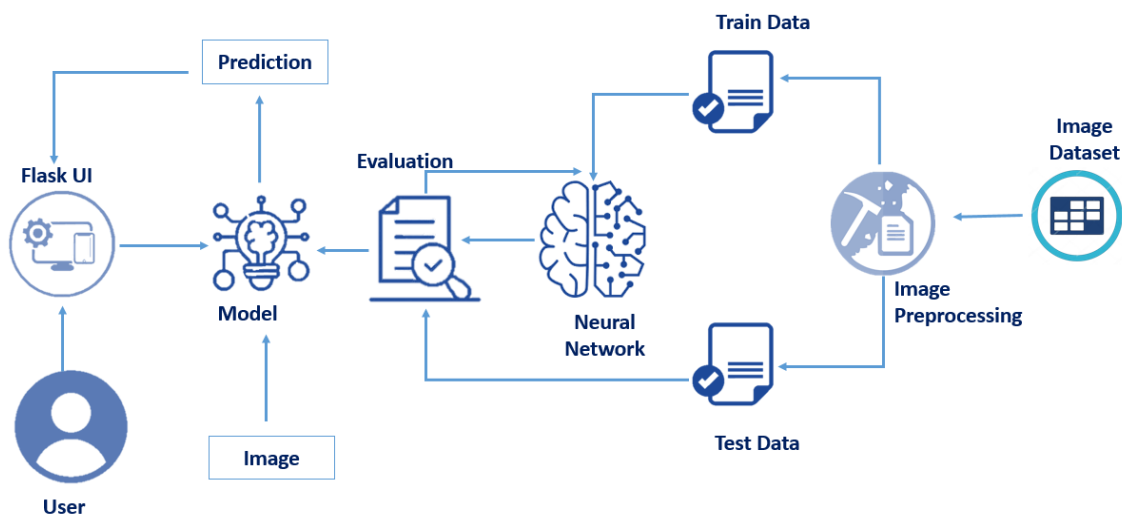
Project Description:

Invasive Ductal Carcinoma (IDC) is the most common subtype of all breast cancers. To assign an aggressiveness grade to a whole mount sample, pathologists typically focus on the regions which contain the IDC. As a result, one of the common pre-processing steps for automatic aggressiveness grading is to delineate the exact regions of IDC inside of a whole mount slide.

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.

Architecture:



Project Flow:

- ❖ Download the dataset.
- ❖ Classify the dataset into train and test sets.
- ❖ Add the neural network layers.
- ❖ Load the trained images and fit the model.
- ❖ Test the model.
- ❖ Save the model and its dependencies.
- ❖ Build a Web application using flask that integrates with the model built.

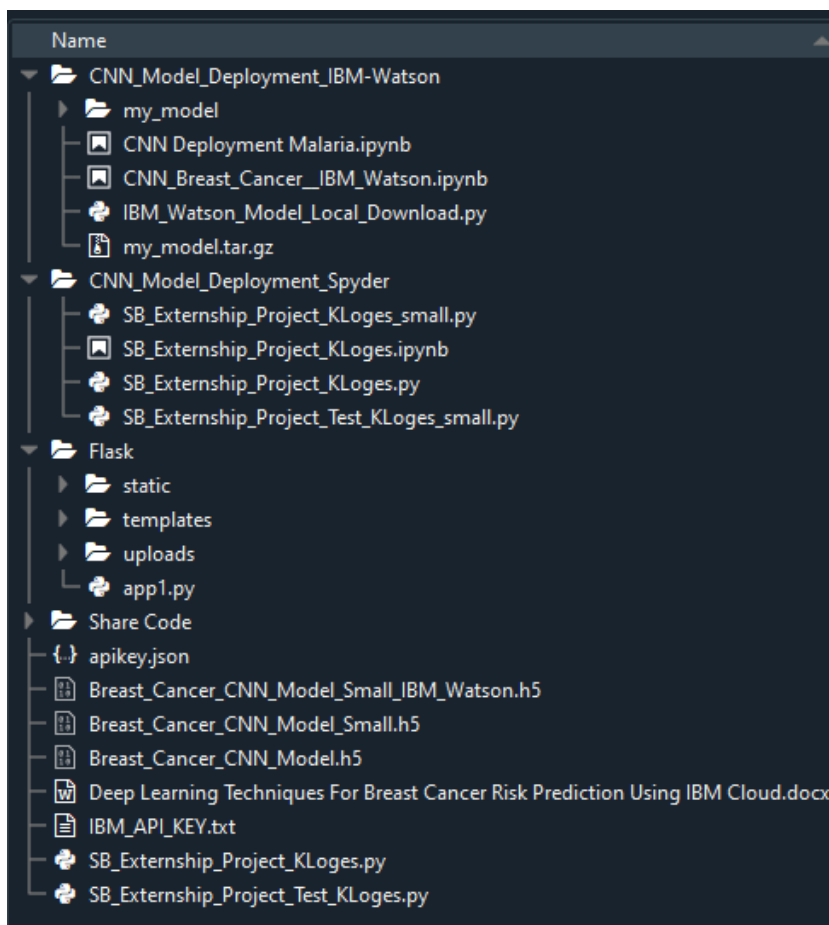
Pre-Requisites:

- ❖ Anaconda Navigator
- ❖ Tensor flow
- ❖ Keras
- ❖ Flask

Prior Knowledge:

- ❖ Supervised and unsupervised learning
- ❖ Convolution Neural Networks
- ❖ Flask

Project Structure:



Data Collection:

- ❖ Create Train and Test folders with each folder having subfolders with images of breast cancer and who don't. You can collect datasets from different open sources like kaggle.com, data.gov, UCI machine learning repository, etc. The folder contains the datasets which can be used for training.
- ❖ Link to download dataset
<https://www.kaggle.com/paultimothymooney/breast-histopathology-images>
- ❖ Dataset Description
The original dataset consisted of 162 whole mount slide images of Breast Cancer (BCa) specimens scanned at 40x. From that, 277,524 patches of size 50 x 50 were extracted (198,738 IDC negative and 78,786 IDC positive). Each patch's file name is of the format: uxYyYclassC.png — > example 10253idx5x1351y1101class0.png . Where u is the patient ID (10253idx5), X is the x-coordinate of where this patch was cropped from, Y is the y-coordinate of where this patch was cropped from, and C indicates the class where 0 is non-IDC and 1 is IDC.

Project Implementation - Spyder

❖ Image Preprocessing

Image Pre-processing includes the following main tasks

- ✓ Import ImageDataGenerator Library.
- ✓ Configure ImageDataGenerator Class.
- ✓ Applying ImageDataGenerator functionality to the trainset and test set.
- ✓ Note: The ImageDataGenerator accepts the original data, randomly transforms it, and returns only the new, transformed data.
- ✓ The dataset images are to be preprocessed before giving it to the model.
- ✓ Output:

```
Found 16178 images belonging to 2 classes.  
Found 3488 images belonging to 2 classes.  
{'0': 0, '1': 1}
```

❖ Model Building

In this milestone, we start building our model by:

- ✓ Initializing the mode
- ✓ Adding Convolution layers
- ✓ Adding Pooling layers
- ✓ Flatten layer
- ✓ Full connection layers which include hidden layers
- ✓ At last, we compile the model with layers we added to complete the neural network structure
- ✓ Fit and Save the model
- ✓ Output:

```
Model: "sequential"  


| Layer (type)                 | Output Shape         | Param #  |
|------------------------------|----------------------|----------|
| conv2d (Conv2D)              | (None, 124, 124, 32) | 2432     |
| max_pooling2d (MaxPooling2D) | (None, 62, 62, 32)   | 0        |
| flatten (Flatten)            | (None, 123008)       | 0        |
| dense (Dense)                | (None, 128)          | 15745152 |
| dense_1 (Dense)              | (None, 2)            | 258      |

  
Total params: 15,747,842  
Trainable params: 15,747,842  
Non-trainable params: 0
```

```
CNN_Model_Deployment_Spyder  
├── Breast_Cancer_CNN_Model_Small.h5
```

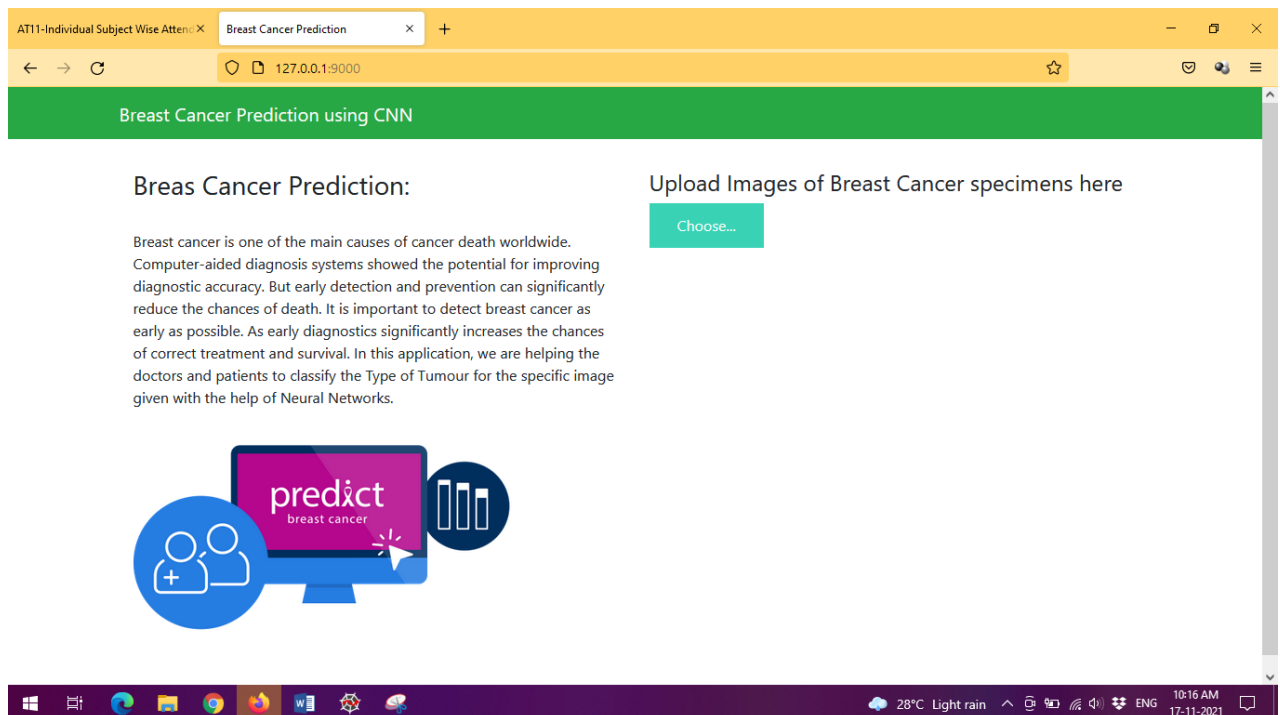
❖ Test the Model

- ✓ The model is to be tested with different images to know if it is predicting correctly.
- ✓ Output:

```
In [3]: y=model.predict(x)
...: pred= np.argmax(y, axis=1)
...: print(pred)
...:
...: index=['Breast Cancer Negative','Breast Cancer
Positive']
...: result=str(index[pred[0]])
...: result
[1]
Out[3]: 'Breast Cancer Positive'
```

Flask Application Building

- ❖ After the model is built, we will be integrating it into a web application so that normal users can also use it. The users need to give the X-ray images to know the predictions.
- ❖ Output:



AT11-Individual Subject Wise Attent...


Breast Cancer Prediction

127.0.0.1:9000

Breast Cancer Prediction using CNN

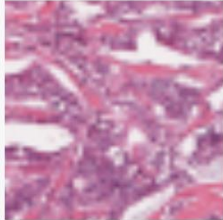
Breas Cancer Prediction:

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. 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.



Upload Images of Breast Cancer specimens here

Choose...



Predict!

28°C Light rain

10:16 AM 17-11-2021

AT11-Individual Subject Wise Attent...


Breast Cancer Prediction

127.0.0.1:9000

Breast Cancer Prediction using CNN

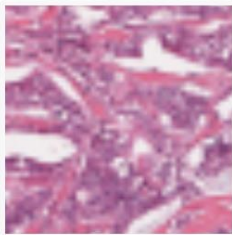
Breas Cancer Prediction:

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. 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.



Upload Images of Breast Cancer specimens here

Choose...



Result: Breast Cancer Positive

28°C Light rain

10:16 AM 17-11-2021

Train the Model on IBM

In this milestone, you will learn how to build Deep Learning Model Using the IBM cloud.

The screenshot shows the IBM Cloud 'Resource list' interface. At the top, there's a search bar and navigation links like 'Catalog', 'Docs', 'Support', and 'Manage'. Below the search bar, there's a 'Create resource' button. The main area displays a table of resources categorized by groups like 'Cloud Foundry services', 'Services and software', 'Storage', 'Network', 'Functions namespaces', and 'Apps'. The 'Services and software' group is expanded, showing 'WatsonMachineLearning' and 'WatsonStudio' as active resources in the Dallas location. The 'Storage' group shows 'CloudObjectStorage' as an active resource in the Global location. The 'Apps' group shows two Node RED instances.

Name	Group	Location	Status	Tags
Cloud Foundry services (0)				
Services and software (2)				
WatsonMachineLearning	Default	Dallas	Active	cpdaas
WatsonStudio	Default	Dallas	Active	cpdaas
Storage (1)				
CloudObjectStorage	Default	Global	Active	cpdaas
Network (0)				
Functions namespaces (0)				
Apps (2)				
Node RED MEWRV 2021-10-28	Default	Global	—	—
Node RED UFPNN 2021-10-28	Default	Global	—	—

The screenshot shows the IBM Watson Studio 'Projects' page. It features a search bar and a 'New project' button. Below, a table lists projects with columns for Name, Role, Storage, Collaborators, Creator, and Date created. The 'Breast_Cancer_Prediction_Logu' project is highlighted, showing it is an Admin role project using COS storage, created by Logeswaran Kanakachalam on Nov 04, 2021. A 'Show more' button is visible below the table.

Name	Role	Storage	Collaborators	Creator	Date created
Breast_Cancer_Prediction_Logu	Admin	COS	LK	Logeswaran Kanakachalam	Nov 04, 2021

The screenshot shows the IBM Watson Studio 'Assets' page for the 'Breast_Cancer_Prediction_Logu' project. It displays a table of assets with columns for Name, Type, Created by, and Last modified. Five assets are listed, including ZIP files for dataset resizing and small versions, and PNG files for specific data classes. Below the assets table, there's a 'Notebooks' section with a table showing a notebook named 'CNN_Breast_Cancer' created by Logeswaran Kanakachalam on Nov 08, 2021, using Python 3.8.

Name	Type	Created by	Last modified
Breast_cancer_Dataset_Resize.zip	Data Asset	Logeswaran Kanakachalam	Nov 04, 2021, 09:13 PM
BreastCancer_Dataset_Small.zip	Data Asset	Logeswaran Kanakachalam	Nov 07, 2021, 01:30 PM
9023_idx5_x201_y1001_class0.png	Data Asset	Logeswaran Kanakachalam	Nov 07, 2021, 03:27 PM
9023_idx5_x1301_y1401_class1.png	Data Asset	Logeswaran Kanakachalam	Nov 08, 2021, 11:38 AM
9023_idx5_x1301_y1401_class1.png	Data Asset	Logeswaran Kanakachalam	Nov 08, 2021, 11:38 AM

Name	Shared	Scheduled	Status	Language	Last editor	Last modified
CNN_Breast_Cancer				Python 3.8	Logeswaran Kanakachalam	Nov 08, 2021

IBM Cloud

Search resources and offerings...

Catalog Docs Support Manage Logeswaran Ka...

Cloud Object Storage

Storage instances Buckets Integrations Endpoints Usage details Service credentials Connections Plan

Storage / CloudObjectStorage /

breastcancerpredictionlogu-donotdelete-pr-hvbo...

Transfers Details Actions...

Objects Configuration Permissions

Objects

Reminder: Objects are uploaded in multiple parts to optimize transfer performance. If an upload is interrupted before the transfer is completed, then the parts of the incomplete object that were uploaded prior to the interruption will count towards billable storage. While the console will alert users to incomplete multipart uploads, it is encouraged to routinely check for and clear out incomplete uploads using the REST API or an SDK. [Learn more](#)

Incomplete Upload

Part of your upload was not completed, incomplete objects will not appear in the object list but will occupy billable space in storage. Review steps of Api-call and helpful information about multi-part and incomplete uploads. [View docs](#)

Prefix filter

Upload

Object name	Size	Last modified
9023_idx5_x1301_y1401_class1.png	6.1 KB	2021-11-08 11:38 AM
9023_idx5_x201_y1001_class0.png	6.1 KB	2021-11-07 3:27 PM
BreastCancer_Dataset_Small.zip	121.5 MB	2021-11-07 1:30 PM
Breast_cancer_Dataset_Resize.zip	1.6 GB	2021-11-04 9:13 PM
notebook/CNN_Breast_Cancer_NozGFstJR.ipynb	174.3 KB	2021-11-08 11:41 AM

Items per page: 10

1-10 of all items

page 1

Drag and drop files (objects) here or click to upload

ibm watson studio

Unlock the power of

Resource list - IBM Cl

Service Details - IBM

IBM Watson Studio

CNN_Breast_Cancer

https://dataplatform.cloud.ibm.com/analytics/notebooks/v2/887H760-a466-4c1b-9f49-82661da1457e/view?projectId=a3d67a23-5e1f-4...

IBM Watson Studio

All

Search

Buy

Logeswaran Kanakachalam...

Projects / Breast_Cancer_Prediction_Logu / CNN_Breast_Cancer

model.add(Dense(units=2, activation="softmax"))

In [18]: model.summary()

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 124, 124, 32)	2432
max_pooling2d (MaxPooling2D)	(None, 62, 62, 32)	0
flatten (Flatten)	(None, 123808)	0
dense (Dense)	(None, 128)	15745152
dense_1 (Dense)	(None, 2)	258

Total params: 15,747,842
Trainable params: 15,747,842
Non-trainable params: 0

In [19]: #configure the learning process
model.compile(loss="categorical_crossentropy", optimizer="adam", metrics=["accuracy"])

In [20]: #fit the model
steps_per_epoch=no. of images in train data/batch_size
#16178/32=506
validation_steps=no. of images in test data/batch_size
#3488/32=109
model.fit(x_train, steps_per_epoch= 506, epochs=30, validation_data=(x_test, validation_steps=109))

0.8624

Out[20]: <tensorflow.python.keras.callbacks.History at 0x7f79c586eac0>

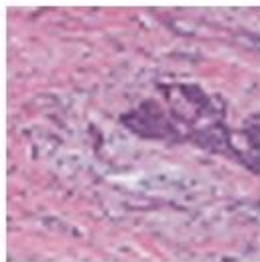
In [21]: model.save("Breast_Cancer_CNN_Model_Small.h5")

In [22]: !tar -zcvf Breast_Cancer_CNN_Model_Small.tgz Breast_Cancer_CNN_Model_Small.h5
Breast_Cancer_CNN_Model_Small.h5

In [23]: ls
Breast_Cancer_CNN_Model_Small.h5 BreastCancer_Dataset_Small/
Breast_Cancer_CNN_Model_Small.tgz

In [41]: im1

Out[41]:



```
In [42]: x=image.img_to_array(im1)
x = np.expand_dims(x,axis = 0)
y=model.predict(x)
pred= np.argmax(y, axis=1)
pred
index=['Breast Cancer Negative','Breast Cancer Positive']
a = index[pred[0]]
print(a)
```

Breast Cancer Positive