ECE1512

Digital Image Processing and Applications

Project Report

Machine Learning on Cancer Prediction

Shuqing Wu - 1000282231

Mahesh Sudhakar - 1004807328

Yuanziwei Li - 1005379387

University of Toronto

March 31, 2019.

Contents

1 Introduction 2

2 Background 3

3 Dataset 4

3.1 Overview 4

3.2 Labels 5

3.3 Splitting the Dataset 6

4 Methodology 7

4.1 Data Pre-processing 7

4.2 Modeling 8

4.2.1 Data Training 8

4.2.2 Testing and Validation 10

5 Results 11

6 Conclusion 14

7 Appendix 15

References 16

1 Introduction

One in every eight deaths in the world is caused by cancer. Breast cancer is the most commonly occurring cancer in women and the second most common cancer overall. There were over 2 million new cases in the year 2018 itself [1]. According to the Canadian Cancer Society, in 2017, about 26,300 women have diagnosed with breast cancer out of which 5,000 women were dead. On average, 14 Canadian women die from breast cancer every day, which is a significant number and it requires immediate attention [2]. Timely earlier diagnosis contributes to the reduction of mortality. However, nowadays, there is still no effective way to prevent the occurrence of breast cancer. Also, the diagnosis of breast cancer lacks efficiency due to the high demand for skilled professional oncologists. Hence, with the development of technology, computer-aided diagnosis(CAD) using Machine Learning has been proposed to improve the accuracy and efficiency of detecting breast cancer.

Recently, there has been an increase in the development of computer-aided cytology and digital pathology. In terms of cancer diagnosis, it is able to abstract abnormal tissue, whether benign or malign, from the patient's breast histology images through image processing techniques. Moreover, artificial intelligence brings us an opportunity to use machine learning algorithms to get better performance of making diagnose. Once a network has been trained from a large amount of labeled dataset, the algorithm comparing with traditional diagnose method can provide more trustworthy result even calculate the possibility. In addition, the quick processing time of the computer system can be regarded as a pre-diagnose method, which saves time for the patient to avoid unnecessary procedure in the hospital. Since patients could get results from the algorithm immediately, it would help patients to make further decisions.

In this project, we mainly consider the automatic detection of breast cancer detection. The core technique is using deep learning network to diagnose breast cancer in Python environment. The dataset is Kaggle's website [3] consisted of breast cancer slide images focusing on a specific type of breast cancer called Invasive Ductal Carcinoma(IDC), the most common of all breast cancer. Section 2 adds preliminaries for our approach including process model and CNN network. Section 3 explains about the dataset chosen and the labels associated with them. Section 4 introduces our method to design our algorithm in order to achieve breast cancer detection basing on Machine Learning technique. Section 5 shows the result of our approach then analyze accuracy and applicability to evaluate this approach. Finally, section 6 draws the conclusion to the whole process.

2 Background

Nowadays, cancer has been prevalent, which is one of the most dangerous diseases that leads to a high mortality rate. In order to detect abnormal issues earlier and offer timely treatment for the patient, we are looking for an approach to enhancing the cancer diagnosis process with this project. The main objective of this project is to scan the image from a part of the human body such as lungs or breast and implement a machine learning algorithm to identify the likelihood of the patient having cancer in that area. We will be taking a set of body scan or microscopic tissue images as inputs, decide how we are going to identify whether the patients have cancer, based on biomedical analysis and label the set with whether they have cancers or not, also train the model with a large amount of labeled images, and finally make use of the algorithm to predict the likelihood of the patient having cancer.

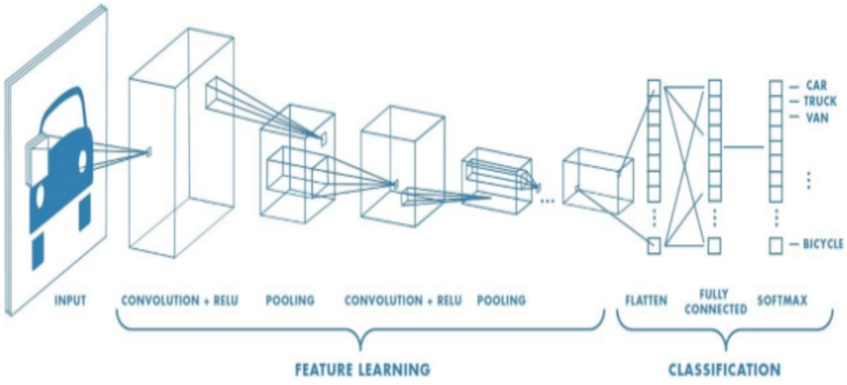

Figure 1: Structure of a CNN

As explained in the article [14], a Comprehensive Guide to Convolutional Neural Networks - the ELI5 way, a Convolutional Neural Network (ConvNet / CNN) is a deep learning algorithm which can take in an image as an input, assign weights and bias to various aspects in the image and differentiate one from the other. The design of convolutional neural network is derived from human brain and Visual Cortex. The network consists of convolutional layer and pooling layer and eventually feeds the results to the classification method. We will be utilizing this network in our application.

3 Dataset

3.1 Overview

Though there are over 8 types of breast cancer found active in women, 'Invasive Ductal Carcinoma (IDC)' is one of the most common subtypes of breast cancer. It affects the rate of cell division in the duct region that carries milk from the lobules to the nipple. Fig. 2 provided by the Canadian Cancer Society [9] shows the position of ducts that are prone to be affected with IDC. The initial stages of IDC are associated only with the ducts and unless they are spread to lobules or any lymph nodes, it is difficult to identify them with CT scan or any other modern scan like PET (Positron Emission Tomography), Ultrasound and MRI scans. So, the conventional technique of malignancy detection using a microscope (pathology) is the only way to ensure early detection of IDC [10], once the patient observes some symptoms.

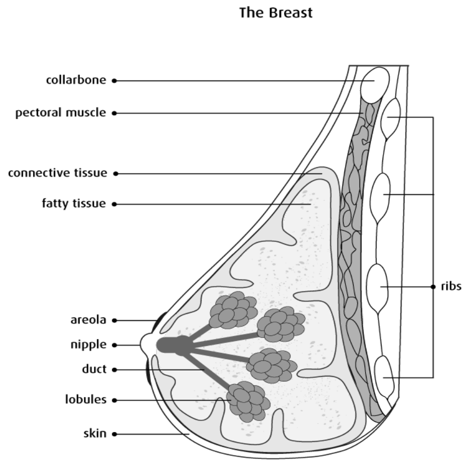

Figure 2: Cross section of breast showing ducts which are prone to IDC

In order to identify differences between the cells and for better visibility, hematoxylin and eosin stains (H&E stains) are added to the specimen before subjected under the microscope. It produces blue, red and violet stains on the tissues, which catalyzes the detection process of the pathologist. As the microscopic images are too small, the initial original dataset contains 162 whole mounted images of Invasive Ductal Carcinoma specimens scanned and zoomed at 40x. Since these original images are massive in terms of spatial dimension to be processed, it has been later curated to 277,524 pathology images of 50x50 dimension each. Then each of these H&E stained images was tested for IDC by various specialized oncologists and radiologists and they are labeled for public use. These processed and labelled images are available in Kaggle, as 'Breast Histopathology Images' by Paul Mooney [3]. The selected dataset contains different 279 folders named with fake patient IDs.

3.2 Labels

Within each folder, we have 2 image subfolders with

- Label 1 - representing “Malignant” (IDC cancer positive)
- Label 0 - representing “Benign” (IDC cancer negative)

Fig. 3 shows samples of pathology images from the dataset, without IDC cancer (Label 0) and Fig. 4 shows sample pathology image with IDC cancer (Label 1)

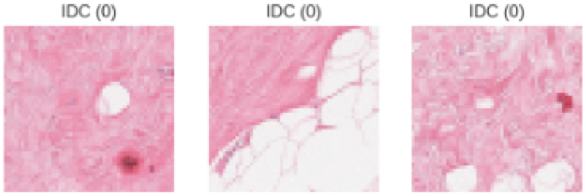

Figure 3: Benign Pathology images

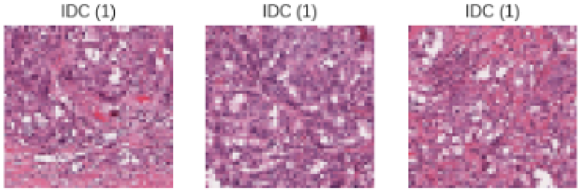

Figure 4: Malignant Pathology images

The violet, red and blue stains in the microscopic pathology images are due to the addition of H&E stains which enables clear differentiation among tissues. Among the 277,524 images in the dataset, 198,738 images are with Label 0 (IDC negative) and 78,786 images are of Label 1 (IDC positive).

Interestingly, each image is labelled with a specific format as ‘p_xA_yB_classC.png’ for easy identification.

For example, consider an image ‘16165_idx5_x1001_y1751_class1.png’. Here,

- 16165_idx5 - denotes the patient ID.
- 1001 - x coordinate of the crop
- 1751 - y coordinate of the crop
- 1 - Label 1 (IDC positive)

This naming convention ensures that all images are easily accessible and to locate them in case of a need. This also has helped a lot during the testing phase of our project, as we can ensure the presence or absence of cancer in an image.

3.3 Splitting the Dataset

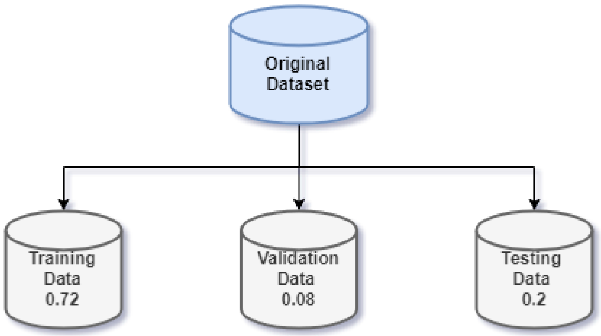

Figure 5: Dataset split configuration

The dataset has been split for training and testing under the usual method. The dataset has been shuffled and 80% of the dataset is used for training and 20% is used for testing. Among the 80%, 10% is used for validation.
Hence, finally,

- 72% of the dataset - Training
- 8% of the dataset - Validation
- 20% of the dataset - Testing

The flowchart image Fig. 5 attached above, illustrates the split among the original dataset. And the Fig. 6 attached below, shows the number of images associated with each dataset split.

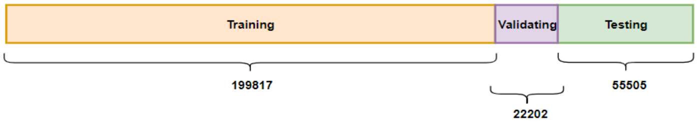

Figure 6: No. of images with each dataset split

4 Methodology

4.1 Data Pre-processing

As the dataset used doesn't have any specific labeling JSON file, the labels are generated as part of our implementation logic. The last value (Label 0 or 1) in the string filename is 'sliced' and curated as labels. These labels are used in the stage of image pre-processing, to divide the training dataset into two folders based on its labels. Later the model is trained on these labels to predict the accuracy in the testing images.

To extract the information from images accurately, we have also applied a few image processing algorithms available such as histogram equalization to balance the intensity of the images and make sure they are consistent for the training model. Histograms of each component (RBG) of the images are calculated and they are plotted to visualize the difference between IDC negative [Fig. 7] and IDC positive images [Fig. 8]. According to the images we tested, it seems to be the blue component of the image that differs in most of the cases. As H&E stained images have blue variation in its result, the histogram of blue component plays an important role in the classification of cancer. As it can be seen in the images attached below, the IDC positive image obtained from the Label 1 folder of the training set, appears to have a wide range of blue color frequency spread out along most of its pixel intensities. But, IDC negative images have a narrow spike in its blue frequency along the 220 – 240 pixel intensities. So, the blue stain of H&E pathology images may be a deciding factor in IDC classification.

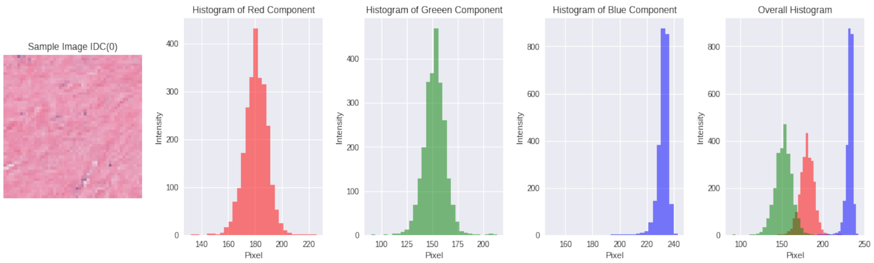

Figure 7: Histogram of each individual component and overall histogram of benign IDC negative image.

During the training phase, for real-time data augmentation and to ensure data regularization, inbuilt Keras *'ImageDataGenerator'* function is used. Initially, each image is normalized, rotated to a particular angle and zoomed, and passed on to the training generator. The dimensions of the image are maintained to be 50x50 along the process. Also, auxiliary augmentation such as height and width shifts, horizontal and vertical flips are done to modify the input images thereby, building the robustness and sensitivity of the training model. These data augmentation techniques would ensure better results for all three main individual steps of our approach, such as feature extraction, feature selection, and classification. The *‘binary cross-entropy’* loss function is utilized during training, as there is only two class available to be classified on, and *’Adagrad’* optimizer is chosen.

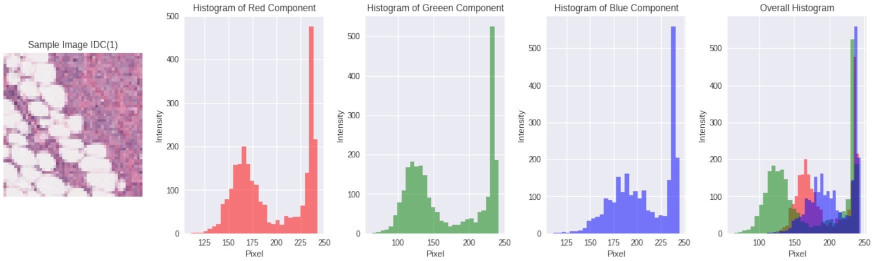

Figure 8: Histogram of each individual component and overall histogram of benign IDC positive image.

4.2 Modeling

4.2.1 Data Training

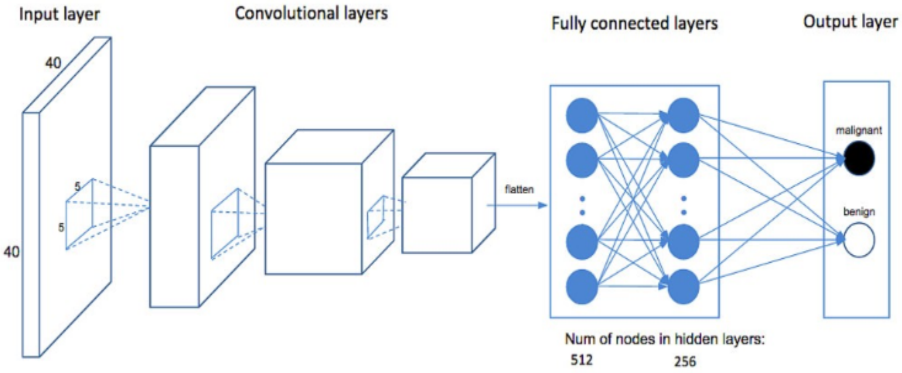

Figure 9: Design of the implemented CNN network

Tool and Environment

We implement a customized CancerNet Model to achieve desired outcome using Keras package for python, which is a high-level neural network API. Packages used include Sequential, BatchNormalization, SeparableConv2D,MaxPooling2D, Activation, Dropout and Dense. They will be explained below based on the Keras documentation referred [15].

Convolutional Neural Network

In this project, Convolutional Neural Network (CNN) is utilized to extract features from the pathology images and performed image classification. CNN has high accuracy and a less free parameter, which are great for large-scale datasets. We construct a CancerNet as our model, and we utilize CNN, RELU activation and pooling layer as the 3 layers of our network.

Input Layer

The model takes the input images and input into the input layer of the network and does the preprocessing step to prepare the images for training.

SeparableConv2D Layer

We make use of separable convolution 2D layer with 3x3 kernel, and 32, 64, 128 output channels.

RELU

Relu is an activation function that's implemented as one layer in the network. The layer outputs x if x is larger than zero, and 0 if x is smaller or equal to zero.

BatchNormalization

In this layer, we normalize each batch activated from the layer before. We apply the transformation function to let mean close to zero, and the standard deviation of activation to 1.

MaxPooling2D

We apply the max pooling operation for spatial data in this layer to reduce the spatial size of convolved feature to decrease the computation power needed for data processing by extracting the most important features.

Dropout

At the end of the network, we apply dropout function to the result to prevent overfitting. Dropout function randomly sets a quarter of input to 0 for each update in training. In this way, we drop some datapoints to balance the results.

Classification Output Layer

In this layer, we first flatten the network and implemented dense and softmax layer. Dense function calculate the activation of (input * weight + bias) as the final result. Then we feed the results in to the final softmax layer.

Softmax Layer

In the classification layer, we have computed the softmax function for the output to get the final classification prediction results. Softmax function is the binary form of logistic regression and calculates the probability of each scenario for classification. The one class with larger probability will be selected.

Fitting the Model

After constructing the model, we fit the model on our training set with adagrad Gradient Descent Optimizer. We first run it with 1 epoch to gather results, then we run it for 40 epochs. The results are shared in the Results section below.

4.2.2 Testing and Validation

After training, we run the fitted model on testing and validation set to generate prediction and accuracy results. We calculate the loss from the test dataset by comparing the generated prediction to the correct label. We also calculate and show the confusion matrix with accuracy, sensitivity, and specificity.

Loss Function

We apply binary cross entropy as our loss function to calculate and minimize our loss from testing and validation.

5 Results

Using all Breast Cancer image data, the diagnose model is built in Python environment basing on learning network is consist of CNN, ReLU, and Pooling layers. In order to calculate computation time, the training and validation process is run both in CPU and GPU (Python3 Google Compute Engine GPU). On CPU mode, each epoch might take around 1.5hr while on GPU model, it takes approximately 105s per epoch. Considering such wide different in computation, finally, we decide to use CPU in small training epoch and apply time-consuming training process with 40 epochs in Google GPU.

First, the model is trained in 1 epoch of dataset. The whole testing dataset is consist of 39711 negative cases(Label 0) and 15794 positive cases(Label 1). Finally, the training loss for one epoch is 0.4198 and accuracy is 0.8240. In terms of validation process, the evaluating result has slightly higher loss and lower accuracy, where the loss is 0.4837 and accuracy is 0.8149. Fig. 10 shows comparison between predicted label from the model and true label from the original dataset. For negative symptom [IDC(0)], there are 32,326 labels resulting from the predicted model match the label from original data while the rest of them result in the opposite label. For positive symptom [IDC(1)], 12,885 test samples recieve correct diagnosis results from predicted model whereas 2,909 images don't perform well.

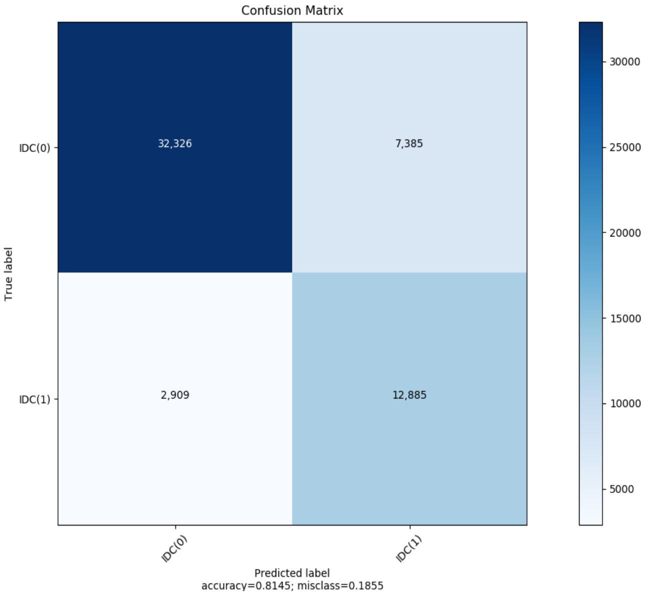

Figure 10: Confusion Matrix - Predicated Label versus True Label for one epoch

In general, Table 1 concludes the performance in one epoch, including three

Classification Report				
Label	Precision	Recall	F-1 Score	Support
0	0.92	0.81	0.86	39711
1	0.64	0.82	0.71	15794
Avg/Total	0.84	0.81	0.82	55505

Table 1: Evaluation Table for 1 Epoch

evaluation scores: precision,recall, and F1-score, a balanced score. For which, the precision for label 0 with 39711 valid labels equals to 0.92 and sensitivity (recall) is 0.81, while the combined score F1-score is 0.86. For label 1 with 15794 valid labels, the precision is much lower but the recall is higher so the F1-score is lower in the end. Therefore, the classification of label 0 performs better than label 1. On average, the F1-score is 0.82, which means that the proposed model is able to capture histology image from the patient and result in diagnosing whether the patient has breast cancer or not.

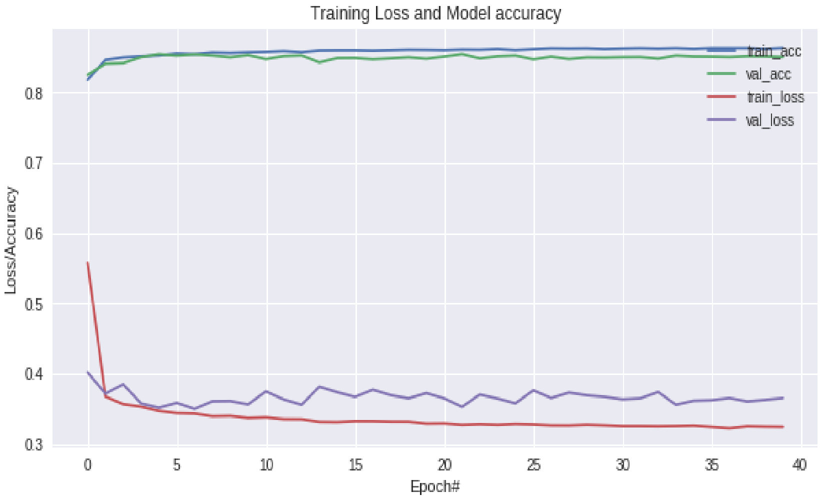

Figure 11: Training and Validation accuracy for 1 to 40 epoch

Then the proposed algorithm with the same dataset is trained and tested in Google Colab GPU environment with 40 epochs. It takes about an hour to compute all training process. Fig. 11 indicates the learning curve in epoch processing. With increase in epoch, the training accuracy is raising dramatically in several steps at start, then growing slowly. However, the validation accuracy grows rapidly in the beginning, then floats around value equals to 0.85. Similarly, during the whole training process, training loss is descending all the time. But validation loss is floating around 0.34 from epoch equals to 10. Therefore, even training accuracy is improving and loss is decreasing by the growth of epoch number, the validation score, including accuracy and loss, tends to float in same value.

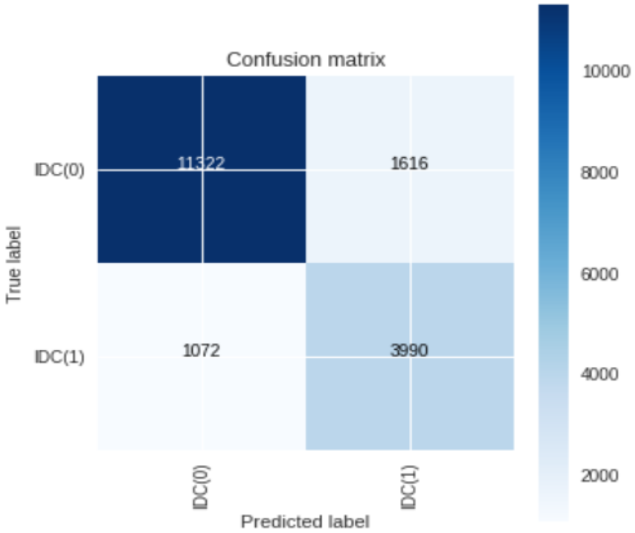

Figure 12: Confusion Matrix - Predicted Label versus True Label for 40 epochs

After that the trained model from 40 epochs is evaluated by testing sample set. Fig. 12 represents the confusion matrix of testing result. In which, the percentage of correct label of label 0 is 87.5% and 78.8% for label 1. Comparing with one epoch, the accuracy for both label 0 and label 1 improve. However, according to the curving change of accuracy shown in Fig. 11, such improved performance could be achieved by training in 10 epochs.

Finally, we randomly pick one image [Fig. 13] from dataset to simulate how user get the result from the trained model and then check the accuracy. The sample and diagnose result show below. The prediction of the sample is negative which is consistent with true label.

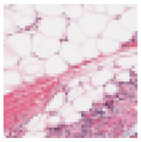

Figure 13: Test image

Prediction: [0]
Expected Result: [0]
Loss: 0.4238
Accuracy: 0.8224
Validation loss: 0.5808
Val Accuracy: 0.8272

Classification Report				
Label	Precision	Recall	F-1 Score	Support
0	1	1	1	1
Avg/Total	1	1	2	1

Table 2: Evaluation Table for a single test image

6 Conclusion

In this report, a novel model is proposed for training a large number of labeled breast cancer histology images from Kaggle's website, then to test and validate on the other set of images to predict cancer.

To be specific, our solution takes Machine Learning algorithm into the medical application and get a relative reliable result to diagnose breast cancer. The learning algorithm through the convolutional layer, activation function ReLU and Pooling layer to connect the image set to their corresponding labels in order to build up a complete network, which results in a reliable network model for breast cancer detection.

Finally, the model results in maximal accuracy around 0.85, where 0.87 accuracy for negative diagnosis and 0.79 accuracy for positive diagnosis. Besides, with training epoch grows from 0 to 10, the accuracy increases rapidly, which indicates that connection in network could be enhanced and improved from training steps. Whereas, after 10 epochs, the accuracy is floating around a certain value and not influenced by the increase in the number of epochs. Therefore, the model has reached its limitation and it is not worthwhile to train the model with more epochs since it is time consuming.

In sum, the proposed approach could detect histology image of the patient and make reliable diagnosis of breast cancer. However, in order to apply such an approach into the real medical field, the model requires more progress to enhance the accuracy in order to guarantee correct diagnosis for the patient. Moreover, this method is limited in specific input image, so we could improve it more in the future to make it fit into another different input type, such as CT image, beside to detect different cancer, such as lung cancer.

7 Appendix

List of Figures

1 Structure of a CNN 3
2 Cross section of breast showing ducts which are prone to IDC . . 4
3 Benign Pathology images 5
4 Malignant Pathology images 5
5 Dataset split configuration 6
6 No. of images with each dataset split 6
7 Histogram of each individual component and overall histogram of benign IDC negative image. 7
8 Histogram of each individual component and overall histogram of benign IDC positive image. 8
9 Design of the implemented CNN network 8
10 Confusion Matrix - Predicated Label versus True Label for one epoch 11
11 Training and Validation accuracy for 1 to 40 epoch 12
12 Confusion Matrix - Predicted Label versus True Label for 40 epochs 13
13 Test image 13

List of Tables

1 Evaluation Table for 1 Epoch 12
2 Evaluation Table for a single test image 14

References

[1] World Cancer Research Fund, https://www.wcrf.org/dietandcancer/cancer-trends/breast-cancer-statistics.Breast Cancer

[2] Canada Cancer Society, http://www.cancer.ca/en/cancer-information/cancer-type/breast/statistics/?region=on.Breast Cancer Statistic

[3] Kaggle breast cancer dataset, https://www.kaggle.com/paultimothymooney/breast-histopathology-images.

[4] Author, F.: Article title. Journal **2**(5), 99–110 (2016)

[5] Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). 10.10007/1234567890

[6] Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)

[7] Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)

[8] Pyimages search algorithm: Breast cancer classification with Keras and Deep Learning, https://www.pyimagesearch.com/2019/02/18/breast-cancer-classification-with-keras-and-deep-learning/

[9] Breast Cancer : Canadian Cancer Society, http://www.cancer.ca/en/cancer-information/cancer-type/breast/breast-cancer/?region=on

[10] Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4977982/

[11] Kadir T, Gleeson F. ‘Lung cancer prediction using machine learning and advanced imaging techniques’. Transl Lung Cancer Res 2018;7(3):304-312.

[12] McWilliams A, Tammemagi MC, Mayo JR, et al.Probability of Cancer in Pulmonary Nodules Detected on First Screening CT. N Engl J Med 2013;369:910-9

[13] Breast cancer dataset: Kinahan, Paul; Muzi, Mark; Bialecki, Brian; Coombs, Laura. (2017). Data from ACRIN-FLT-Breast. The Cancer Imaging Archive. http://doi.org/10.7937/K9/TCIA.2017.ol20zmxg

[14] A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[15] Keras Documentation: https://keras.io/

University of Toronto

ECE1512
Group Final Report

Title:	Machine Learning on Cancer Prediction	
Project I.D.#:	6	
Team members: (Select one member to be the main contact. Mark with '*')	Name:	Email:
	Shuqing Wu	shuqing.wu@mail.utoronto.ca
	Mahesh Sudhakar	mahesh.sudhakar@mail.utoronto.ca
	Yuanziwei Li	weiziyuan.li@mail.utoronto.ca.
Supervisor:		
Submission Date:	April 1st, 2019.	
Additional Comments		

Group Final Report Attribution Table

This table should be filled out to accurately reflect who contributed to each section of the report and what they contributed. Provide a **column** for each student, a **row** for each major section of the report, and the appropriate codes (e.g. 'RD, MR') in each of the necessary **cells** in the table. You may expand the table, inserting rows as needed, but you should not require more than two pages. The original completed and signed form must be included in the hardcopies of the final report. Please make a copy of it for your own reference.

Section	Student Names			
	Shuqing	Mahesh	Yuanziwei	
Introduction	RS/RD/MR/ET		RS/RD/MR/ET	
Background	RS/RD/MR/ET			
Dataset		RS/RD/MR/ET		
Methodology – Preprocessing		RS/RD/MR/ET		
Methodology – Modeling	RS/RD/MR/ET			
Results			RS/RD/MR/ET	
Conclusion			RS/RD/MR/ET	
All	FP/CM	FP/CM	FP/CM	

Abbreviation Codes:

Fill in abbreviations for roles for each of the required content elements. You do not have to fill in every cell. The "**All**" row refers to the complete report and should indicate who was responsible for the final compilation and final read through of the completed document.

RS – responsible for research of information
RD – wrote the first draft
MR – responsible for major revision
ET – edited for grammar, spelling, and expression
OR – other
"All" row abbreviations:
FP – final read through of complete document for flow and consistency
CM – responsible for compiling the elements into the complete document
OR - other
If you put OR (other) in a cell please put it in as OR1, OR2, etc. Explain briefly below the role referred to:
OR1: enter brief description here
OR2: enter brief description here

Signatures

By signing below, you verify that you have read the attribution table and agree that it accurately reflects your contribution to this document.

Name SHUQING WU Signature Date: Apr 1, 2019.

Name MAHESH SUDHAKAR Signature Date: 1/4/19

Name Yuanziwei Li Signature Date: 1/4/2019

Name Signature Date: