Early Stage Detection and Classification of Colon Cancer using Deep Learning and Explainable AI on Histopathological Images

by

Mainul Hossain
15341003
Shataddru Shyan Haque
21241079
Humayun Ahmed
17101358
Hossain Al Mahdi
17201084
Ankan Aich
18101445

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
BRAC University
January 2022

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

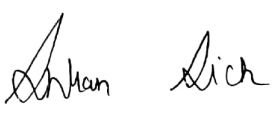
Student's Full Name & Signature:

Mainul Hossain
15341003

Shataddru Shyan Haque
21241079

Humayun Ahmed
17101358

Hossain Al Mahdi
17201084



Ankan Aich
18101445

Approval

The thesis titled "Early Stage Detection and Classification of Colon Cancer using Deep Learning and Explainable AI on Histopathological Images" submitted by

1. Mainul Hossain (15341003)
2. Shataddru Shyan Haque (21241079)
3. Humayun Ahmed (17101358)
4. Hossain Al Mahdi (17201084)
5. Ankan Aich (18101445)

Of Fall 2021, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 20, 2022.

Examining Committee:

Supervisor:

(Member)

Md Tanzim Reza
Lecturer
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:
(Member)

Zavid Parvez
Digitally signed by Zavid Parvez
DN: cn=Zavid Parvez, o=Brac University, ou=CSE, email=zavid.parvez@bracu.ac.bd, c=US
Date: 2022.01.20 23:44:56 +11'00'

Dr. Mohammad Zavid Parvez
Assistant Professor (Former)
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Associate Professor
Department
BRAC University

Head of Department:
(Chairperson)

Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
BRAC University

Ethics Statement

We the individuals, therefore and truly pronounce that this thesis report has been done based on the findings of our extensive research. This report correctly notes and cites all of the materials that were utilized. This research work, not one or the other in full nor any portion has never been submitted by any other individual to another university or any institution for the grant of any degree or for any other reason.

Abstract

Colon cancer is one the most prominent and daunting life threatening illnesses in the world. Histopathological diagnosis is one of the most important factors in determining cancer type. The current study aims to create a computer-aided diagnosis system for differentiating tissue cells, benign colon tissues, and adenocarcinomas tissues of the colon, using convolutional neural networks and digital pathology images for such tumors. As a result, in the coming years, artificial intelligence will be a promising technology. The LC25000 dataset, which included 5000 photographs for each class, produced a total of 25000 digital images for lung and colonic cancer cells, as well as healthy cells. The photos of lung cancer were not included in our study because it was primarily focused on colon cancer. To categorize and classify the histopathological slides of adenocarcinomas and benign cells in the colon, a Convolutional neural network architecture was implemented. We also explored optimization techniques such as Explainable AI techniques, Lime and DeepLift to better understand the reasoning behind the decision the model arrived at. This allowed us to better understand and optimize our models for a more consistent accurate classification. Diagnosis validity of greater than 94% was obtained for colon distinguishing adenocarcinoma and benign colonic cells

Keywords: Colon Cancer, Deep Learning, CNN, Image Classification, Whole Slide Images, Histopathological Images, Explainable AI, Optimization algorithms, Lime.

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.
Secondly, to our advisor and co-advisor Md Tanzim Reza sir and Dr. Mohammad Zavid Parvez sir for their guidance, kind support and advice in our work. They helped us whenever we needed help.
And finally to our parents without their throughout support it may not be possible. With their kind effort and support and prayer we are now on the verge of our graduation.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$AdaGrad$ Adaptive Gradient Algorithm

AI Artificial Intelligence

AUC Area under the ROC Curve

$CFCMC$ Cumulative Fuzzy Class Membership Criterion

CNN Convolutional Neural Network

$LIME$ Local Interpretable Model-Agnostic Explanation

$LSTM$ Long Short-Term Memory

NEP Neighbour Ensemble Predictor

$ReLU$ Rectified Linear Activation Function

$ResNet$ Residual Network

$RMSProp$ Root Mean Squared Propagation

$RMSProp$ Root Mean Squared Propagation

RNN Recurrent Neural Network

$SC-CNN$ Spatially Constrained Convolutional Neural Network

$SR-CNN$ Super Resolution Convolutional Neural Network

$SSPP$ Single Sample Per Person

$TCGA$ The Cancer Genome Atlas

VGG Visual Geometry Group (UK)

WSI Whole Slide Images

$X-CFCMC$ Explainable Cumulative Fuzzy Class Membership Criterion

XAI Explainable Artificial Intelligence

Chapter 1

Introduction

In both developed and developing countries, cancer is a leading cause of death. Aside from the fact that it causes death, the related medical costs are substantially significant and have an impact on both public and private healthcare systems, impacting both the government and the general public. As a result of risk factor reduction efforts (e.g., smoking, obesity, lack of exercise) and treatment improvements, the mortality rate in high-income countries has remained consistent or even decreased, according to a study conducted by Torre et al. [20].

1.1 Background

Colorectal cancer being the third most prevalent cancer in males and the second leading cause of cancer in women, with over 1.8 million confirmed cases in 2018, according to Siegel et al. (2020) [30]. Despite the fact that colon cancer has a high occurrence rate, it is difficult to detect it at an early stage. However, a variety of screening procedures are employed to detect said cancer, but they are not entirely reliable in providing accurate prognosis for a patient's health and future survival. Over the years, significant study and progress has been conducted in the area of colon cancer risk reduction, offering us with motivation to continue combating it.

Several targeted treatments based on molecular research necessitate the gathering and processing of tumor tissue from paraffin blocks. Tissues are then taken from the blocks and examined under a microscope before being digitally processed into Whole-Slide Images. Furthermore, by acting as a diagnostic tool, a computerized solution might possibly save pathologists time and work while also reducing diagnostic uncertainty. As a result, we can have two types of diagnosis: machine findings and doctor diagnoses, which will improve the accuracy of the investigations [31].

When it comes to the effective diagnosis and treatment of many types of cancer, histopathology, or the microscopic inspection of aberrant tissue, is crucial. Researchers and pathologists can now extract relevant information from whole-slide images (WSIs) of cancer tissue more easily, thanks to advances in digital microscopy and the use of deep learning technologies [12].

Deep Learning has made great progress in recent years as the backbone of Machine Learning, allowing it to be effectively applied to a number of applications such as

image processing, audio processing, and language processing. Deep learning could be used to handle medical image analysis, such as magnetic resonance imaging, whole slide image analysis, computed tomography, and so on, according to recent improvements [15]. Researchers may now evaluate the practicality of employing Machine Learning and Deep Learning algorithms to improve the efficiency and reliability of histological examination due to the public availability of digital pathology datasets. The primary goal of our research is to investigate the fact, if features extracted from histopathological images can be used to distinguish colon cancer cells from healthy cells using whole slide images of both cancerous and non-cancerous cells, by implementing Deep Learning models like VGG-16, ResNet50, Inceptionv3, EfficientNet, and Machine Learning techniques like Support Vector Machines, K Nearest Neighbor, and Adam Optimization etc. as well as implementing a new approach towards using explainable AI for the better understanding of how these algorithms are working so that better optimization might be done to increase the accuracy of our predictions.

1.2 Problem Statement

In today's medical society, there are various methods for screening and diagnosing colon cancer, such as colonoscopy, computed tomography, fecal occult blood test, sigmoidoscopy, and so on. However, all of these screening tests have a very low margin of early stage detection and death rate reduction, with some having a reduction rate of less than 30% and 18% when done yearly and every other year respectively. Bangladesh has 13 to 15 lakh cancer patients, according to Hussain, S. A., and Sullivan, R. (2013) [3], with roughly 2 lakh new cases identified each year and as the population grows and people live longer, the number of cancer patients is expected to rise in the coming years. Increased cancer prevalence is made inevitable when the factor of aging is considered.

Colorectal cancer, being the third most prevalent cancer diagnosed worldwide and the fourth largest cause of cancer mortality, making it a serious public health concern. Exacerbations due to differential exposure to risk factors, the development and implementation of screening, and access to appropriate treatment choices, there is significant variance over time across various geographic regions. Furthermore, economic status may explain a sizable amount of the inequalities. Although colorectal cancer is still mostly a disease of the industrialized world, occurrence rates in developing countries are increasing [8].

Despite breakthroughs in medical knowledge and technology, the risk of colon cancer is still at an all-time high, and there are few early detection procedures available.The delayed appearance of colonic cancer symptoms is one of the main causes.However, while the symptoms may be late, the disease can be diagnosed in its early stages if the colonic lining cells are inspected under a microscope.Even in the later stages of the disease, it is difficult for a pathologist to identify healthy tissue cells from malignant tissue cells.

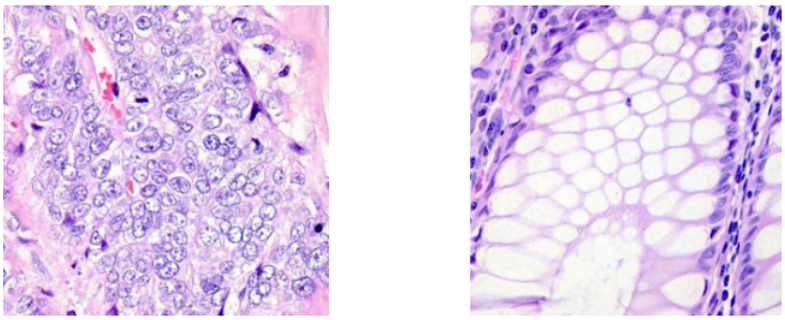


Figure 1.1: Colon tissue with Adenocarcinomas (Left) and Benign colon tissue (Right)

Despite the fact that there are treatments for colonic cancer, the late onset of symptoms, as well as pathologists' difficulty distinguishing healthy from cancerous colonic tissue, causes a drastic reduction in a human's chance of survival. As a result, our goal for this research is to detect and distinguish healthy from cancerous cells in the early stages using whole slide images processed through Deep Learning Models.

Chapter 2

Literature Review and Research Objectives

2.1 Literature Review

This research aims to use Whole Slide Images of Cancerous and Benign Colon Tissue to detect and categorize colon cancer in its early stages. On the Lung and Colon Cancer Histopathological Image Dataset, (LC25000) [23], an ensemble of Convolutional Neural Networks will be utilized to demonstrate classification performance and accuracy. substantial amounts of research for the study of Colon Cancer have used Hybrid Deep Learning Networks, an amalgamation of many CNN networks, with improved accuracy of up to 90% in most cases. Earnest Paul Ijjina, Chalavadi Krishna Mohan, Applied Soft Computing, Volume 46 [11], where a hybrid deep learning system is used For distinguishing human behaviors in videos, CNN classifiers were utilized, resulting in a high recognition accuracy of 99.68 %. Applying both Machine Learning and Deep Learning algorithms to get a high accuracy score for the categorization of colon cancer in various techniques in research cases of colon cancer has had a lot of success. Furthermore, several efforts have been attempted to develop a model that can be used to develop accurate early detection systems for colon cancer.

Shapcott et al. (2019) [24] took a different strategy, implementing a double CNN (Convolutional Neural Network) model to first locate the cells and then identify the type of cells found. WSIs (Whole Slide Images) are used as input in this method where the first CNN uses them directly and the output is then passed on to the second CNN, which categorizes or classifies it. The WSI was tiled with square patches that were 500 pixels wide (20X) and a total of 900 patches were employed, each containing a large amount of tissue. By sampling enough patches without surrendering too much precision, the obvious issue of high computing cost was mitigated. The algorithms were inspired by Sirinukunwattana et al. (2016) [17], who employed histology pictures to detect and classify the nuclei in colon cancer cells, as well as their detection.To achieve much better accuracy in labeling any object, a locally sensitive deep learning approach was used to determine the distance from the nucleus where the center of the said object is included when calculating the probability map for detection, in addition to a weighted array of local indicators for a classifier label. A spatially constrained CNN, or SC-CNN, may be utilized to predict the probability of

a pixel having been located in the nucleus's center . This CNN variation incorporates the principles consisting of a parameterization layer and a spatially limited level for spatial regression. In addition to a normal softmax CNN, the Neighbour Ensemble Predictor (NEP) was used for classification. A total of 29,756 nuclei were identified, with 22,444 having an associated cell type designation such as fibroblast, epithelial, inflammatory, and others. This softmax CNN and NEP technique achieves a multi-class AUC of 0.917 and an average weighted score of 0.784.Furthermore, a softmax CNN and single sample per person (SSPP) combination was used, which had a very small but lower accuracy than the softmax CNN and NEP technique, which was used alongside SC-CNN and SR-CNN. Similarly, Shapcott et al.(2019) [24] developed a cell identification algorithm based on systematic random cell sampling. A cell's profile was created utilizing 5 TCGA diagnostic pictures and 1,500 cells that were manually marked by a pathologist. The detection and classification accuracy of these patches was 65%. This method is especially useful when the regions of interest are sparse and concentrated in space.

Mangal et al. (2020) [28] used a different Deep Learning technique based on Borkowski et al. [23] where Microscopic LC25000 Dataset of whole slide Lung and colon images are used. The dataset is divided into five categories, each containing 5000 images: lung adenocarcinomas, lung squamous cell carcinomas, lung benign, colon adenocarcinomas, and colon benign.The initial dataset contains 500 colon pictures with 1024x768 pixel dimensions that were then translated to squares with 768x768 pixel sizes. Using rotation and flips, Augmentor was utilized to enhance the dataset to 25000 frames. Originally, data was randomly collected into 4500 and 500 data points for each class's training sets and test sets, accordingly. The photos were then downsized to 150x150 pixels and subjected to randomized shear, zoom transformation, and picture normalization. Deep Learning's capacity to expose abstract level information and then plunge to extract fundamental semantic features, as indicated by Mangal et al. (2020) [28], is among the important attributes required for its development. CNNs have recently proven to be a useful tool for classifying and analyzing image-based extraction of features due to its capacity to stack several trainable layers alongside a trained classifier to generate feature maps from a specified data input feed. The CNN architecture is made up of three layers: the convolution layer, the max pooling layer, and the dense layers. For their CNN architecture, the authors employed the following architecture and training technique: first, information is provided to the first convolution layer via an input layer, with images of 150x150 pixels and color channel 3 for RGB. The Convolution layer convolved the input sequence using adaptable filters to understand the geo spatial structure of the data; this method comprises three convolution layers, each one with a filter size of 3x3, a stride of 2, and constant padding. That the very first layer consists of 32 filters, which are proceeded by two tiers of 64 filters each, each having a Gaussian distribution as such an initialization. ReLU activation is also utilized in nonlinear procedures to improve performance.The max pooling layer, also known as the pooling process, is used to downsample the convolution layer's output images where each convolution layer is followed by one pooling layer with a pooling size of 2 as well as padding set to valid. The max pooling approach is used by all pooling levels. The Flatten layer is then applied to turn the convolution layer's output into a 1D tensor, which will then be utilized to link a dense layer. The Dense

layer provides a vector output by treating the input as a simple vector. This model has two dense layers, the first of which has 512 neurons and the second of which, depending on the input class, contains 3 for lung cancer 2 neurons colon cancer, correspondingly. The very last fully connected layer's output was activated as a result of Softmax activation. Lastly To prevent model layer overfitting, a dropout layer is added across fully connected layers that periodically eliminates neurons both from visible and hidden layers. The RMSprop approach using backpropagation has been used to calculate gradients, and a minimum batch size of 32 was employed to adjust network weights, with a beginning learning rate of 10-4 and also in tandem with $\rho = 0.9$ and $\varepsilon = 1\mathrm{e} - 7$.. Categorical cross entropy loss 2 is used to ensure that the model's performance remains constant during the training phase. The CNN model was trained for 100 iterations on data that was divided into three sets, each comprising 80-10-10 of data divided into training, testing, and validation sets. The accuracy of approximately 96.9503% was achieved on the training set with a loss of 7.9340%, and an accuracy of 96.6110% was achieved on the validation set with a loss of 9.7141%, demonstrating the superiority of Deep Learning Models for classification over traditional Machine Learning techniques.

In Iizuka et al. (2020) [27],Gastric and colonic epithelial tumors were detected by histological categorization. Researchers utilized 4,128 stomach WSIs and 4,036 colon WSIs from Japan's Hiroshima University Hospital. The remainder of the sample was donated by Haradoi Hospital, which contained 500 stomach WSIs 500 colon WSIs (Fukuoka, Japan). The Hiroshima University Hospital subjects were randomly separated on a WSI level to obtain 500 WSIs for each organ's test set, with the remainder used for training and validation (5%). Rather than being used for training, the Haradoi Hospital cohorts were used as independent test sets. Only few surgical excision cases were included in the colon training set (5%). Furthermore, test sets were obtained from the publicly accessible repository of impartial stomach and colon surgical excision patients maintained by The Cancer Genome Atlas (TCGA) initiative. The inception-v3 network, which has been designed from the ground up, was the core model architecture. The models were trained at a magnification of 20X on 512 by 512 pixel tiles chosen at random from the training sets. Adenocarcinoma, adenoma, or non-neoplastic were the labels assigned to each tile. During training, other data augmentations, such as tile rotations and color changes, were applied to improve the network's robustness and regularization. During inference, the neutral connection are being used in a sliding window technique with input tiles of 512 $\times$ 512 pixels as well as a fixed rate less than the input tile size. Because all tiles must be categorized in order to receive a WSI categorization, a sliding window was used. Heatmaps with smaller strides have prolonged inference periods, while heatmaps with larger strides have reduced inference periods. To construct WSI classifications, max-pooling was employed, which delivers a WSI the label with the maximum probability from all tiles, and an RNN model, which had been trained to include data from the many tiles utilizing deep CNN features as input. On its own test set, the colon model was tested using two aggression methods: max-pooling (MP-aggr) and RNN (RNN-aggr). Performance AUC (Area under the curve) scores were collected and documented. To avoid generalization, the authors also evaluated the model on a different medical institution's test set, resulting in an entirely distinct test set. By omitting the last fully-connected classification layer, this trained inception-v3

network could be used as a feature extractor. The inception-v3 feature extractor generates a 715 feature vector with a depth multiplier of 0.35. A sliding window with such a width of 256 by 256 pixels was used to extract tiles from a WSI. To obtain a WSI classification, every one of the tiles from a particular WSI must be validated during inference. Any length sequence can be used to train an RNN model to produce a single output. We extracted an arbitrary figure of tiles from either the tissue regions for each slide input them it into the RNN model. To avoid relying on tile input order, the order of the attributes of the tiles was randomly changed at each level of training. We employed an RNN with two LSTM layers and 128 hidden state approximations, each with 30 levels. The model has been trained using stochastic gradient descent with a batch size of one. The classifier was tested for 50 epochs at 0.001 learning rate and 1e-6 decay. The final model was picked based on the model that performed the best in the 5% validation subgroup. TensorFlow31 was used to generate and train the models. The AUCs were calculated in Python using the scikit-learn package32 and visualized using matplotlib33. The bootstrap method34 with 1000 iterations was used to obtain the 95 percent CIs of the AUCs. To compare the AUCs of two correlated ROC curves, the two-tailed DeLong's test35 was utilized. The partnered two-sided student t-test was used to compare log loss pairs. A comparison analysis was performed to determine the relationship between pathologists' years of expertise and accuracy. A Student t-test was used to examine pathologists' and medical students' diagnosis accuracy.

Sabol et al.(2020) [29], developed a research based on histopathological images and their evaluations of Hematoxylin and Eosin (HE) stained tissue sections. That is one of the aspects of their research and the goal is to make the results human friendly. Cumulative Fuzzy Class Membership Criterion (CFCMC) formulates its decision through three processes: symbolic explanation of the probability of misclassification, showing the training samples for the results and training samples for class conflicts. Dataset used for training was 5000 small tiles with the 150× 150 pixel labeled with one of eight tissue classes. CFCMC is used to classify the training set. Along with the tissue classes tumor epithelium, simple stroma , complex stroma, immune cells , debris , normal mucosal glands, adipose tissue, background 10 non-annotated WSI or while slide images of the tissue were used. 625 tiles consist of the classes and class balanced data was used. Moreover, the unexplainable portion of the result is generated by the CNN while the X-CFCMC generates the explainable one. The explainable part consisted of semantic explanation and visualization for the results. Probability distribution and prediction is the result of the CNN while the explainable AI part shows explanations for the CFCMC.A CNN or convoluted neural network was used as a feature extractor in this case to improve the accuracy of the CFCMC classifier . Since CNN will compress the data, backtracking for an explanation was difficult for them but the goal of getting an idea of the classifiability of data was achieved. Furthermore, the CFCMC classifier was able to provide visual explanation for the results extracted from the classification of the WSIs of colorectal cancer. They further reviewed the results by 14 pathologists of their XAI(Explainable Artificial Intelligence) to assess their results. Their results consisted of resnet50 giving 93.80% for the CNN model and 92.78% accuracies for the CFCMC model which are the highest amongst the 11 architectures they have used.

2.2 Research Objectives

The goal of this study is to create a Deep Learning model for detecting early stage colon cancer utilizing histopathological images that will be processed using image Classification to extract features from WSI (Whole Slide Images) to train our model. We intend to create a model with a high accuracy prediction of the prognosis of a patient with colon cancer in the early stages using the dataset LC25000 [23] paired with Deep Learning Algorithm CNN (Convolutional Neural Networks) with further optimization and understanding of how the algorithms are behaving we intend to pair the algorithms with Explainable AI libraries such as LIME in python, which will give us a better insight and therefore control in tweaking the parameters of our model in order to find the optimum model.

- Colon cancer classification using histopathological image and feeding them as input to CNN models employing architectures such as VGG-16, ResNet50 and Inceptionv3.
- For image Classification input Whole Slide images of cancerous and non cancerous colonic cells will be used. The LC25000 [23] Dataset will be utilized with images divided into two groups: Adenocarcinoma of the colon (colon aca) and benign colon tissue (colon n).
- A Convolutional layer is used to apply trainable filters to convolve the image features. Max pooling layer will be used to downsample output images from the convolution layer and Dense layer treats the input as a simple vector and creates a vector output. Softmax activation function will be used in the last fully connected layer to obtain final output from the model along with a dropout layer that will help in preventing overfitting.
- Explainable AI algorithm such as LIME, will be used to observe the behavior of the Algorithm so that further improvements and optimizations can be made in order to achieve optimum accuracy and performance
- Final output should allow us to perceive and give a probabilistic estimation on the chances of whether a generalized outside colonic tissue samples contains traces of colon Adenocarcinomas or not

Chapter 3

Methodology and Work Plan

3.1 Dataset

Lung and Colon Cancer Histopathological Image Dataset (LC25000) Borkowski et al.[23] LC25000 Dataset comprises lung and colon images of microscopic degree . The dataset has been split into five categories comprising 500 images. These images have the cells namely:lung adenocarcinomas, lung squamous cell carcinomas, lung benign, colon adenocarcinomas, and colon benign. As the focus of our research is on colon cancer, no lung cancer tissue images were used.The original dataset only comprised 750 photos of the lung and 500 photographs of the colonic cells with pixel sizes of 1024x768, which were then transformed into squares of 768x768 pixels. With the use of rotation and flips, the dataset was increased to 25 thousand images with the help of an augmentor. It is preprocessed before being fed additional data. Sampling of the data is done as 4000 (80%) and 1000 (20%) data points for the training and test sets, respectively, with 50% of the test set serving as a validation set. Random sampling is implemented on each class for the validation set. Furthermore, photos were downsized to 80x80 pixels, with some randomized shear, zoom transformation, and image normalization.

3.2 Convolution Neural Network

To classify our chosen dataset we used the following base model CNN architecture of our own making with the following layers and parameters. Since the beginning of the year 1990, a variety of CNN models were gradually introduced. AlexNet [10] is among the Deep CNNs that helped popularize convolutional networks in computer vision. With almost 60 million parameters and Relu as either a non-linearity function, and the strategy of overlapping Max Pooling and overlaying the convolutional layers, this network was more sophisticated than others. Furthermore, VggNet [5] or Visual Geometry Group has been recognized for its great performance due to its extremely deep architecture. This network used 33 much smaller filters in each convolutional layer. Inception or GoogLeNet [7] offer a significant benefit in terms of parameter reduction since they use the average pooling concept rather than completely linked levels with convolutional layers. Finally, the Residual Network [9] makes a significant contribution by employing batch normalization and the usage of bypass connections to push forth the networks for training higher designs. Additionally, the CNN architecture overall enables a number of fundamental operations.

3.2.1 Input Layer

This layer imports input and sends it to first convolution layer. In this case, the data is an 80x80 pixel image having 3 RGB color channels.

3.2.2 Convolutional Layer

Every convolutional layer requires the learning of a collection of filters and parameters. Furthermore, the dimensions of these filters are always less than the input filter. Each one of the filters was compounded over the raw image channels to build an activation map. The system may learn filters that respond optimally to a specific segment of the input since the convolutional layers are locally connected. Furthermore, the activation map was created using convolutional processes between the filter, the input, and the filter parameters. The following equations summarize an input image and generate an output image by conducting convolution across k channels.[6]

$$A_o^{(m)} = g_m \left(\sum_k W_{ok}^{(m)} * A_k^{(m-1)} + b_o^{(m)} \right)$$

$$W_{ok} * A_k[s,t] = \sum_{p,q} A_k[s+p, t+q] W_{ok}[P-1-p, Q-1-q]$$

3.2.3 Pooling Layer

The linear layer is constructed upon that neuron, which would be the primary processing unit of the brain [26]. The neuron works logically by accepting input impulses and then generating output signals. Additionally, the linear layer seems to be a schematic representation of a group of neurons' dendrites that are all linked to the very same inputs. To mimic the 1-0 impulse absorbed as an activation function, the SoftMax activation function was used. An activation function, on either hand, is indeed the identity function that generates true values.

$$y = A \cdot x + b$$

$$y_i = \sum_{j=1}^{i} (A_{i,j} x_j) + b_i$$

The layer has to have a bias parameter, b which is depicted in the preceding equation.

3.2.4 Linear or Fully Connected Layers

The linear layer is built on the neuron, which is the brain's primary processing unit [26]. The neuron functions in a logical manner, receiving input impulses and then producing output signals. Furthermore, the linear layer is a schematic version of a collection of neurons' dendrites connected to the same inputs. The SoftMax activation function was utilized to replicate the 1-0 impulse carried away as an activation function. The activation function, on the other hand, is the identity function that produces the real values.

$$y = A \cdot x + b$$

$$y_i = \sum_{j=1}^{i} (A_{i,j} x_j) + b_i$$

The layer has a bias parameter, b which is shown in the Equation above.

3.3 Activation Functions

Any non-convex function, and the outcome of non-linear activation functions, can be predicted using neural networks. Those functions also take a vector as an input and perform a defined point-wise action. There are three types of activation functions: binary, linear, and non-linear. For deep learning, we exclusively employ non-linear activation functions. To control the gradient learning rate of deep learning models, various activation functions were created.

3.3.1 Softmax Activation Function

The logistic function would be a multidimensional variation of the softmax function. In multinomial logistic regression, it is widely used as the last activation function of such a neural network to standardize the result of a system to a probability distribution across predicted classes. The softmax function takes as input a vector z of K real numbers then normalizes everything into a probability distribution having K probabilities proportionate to the input values' exponentials[22].

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{for } i = 1, \ldots, K \text{ and } \mathbf{z} = (z_1, \ldots, z_K) \in R^K$$

The Equation above is the standard form of the softmax function when when K is greater than one by the formula.

3.3.2 Rectified Linear Unit (ReLU)

The below equation is a easy form of the ReLU function

$$y = \max(0, x)$$

ReLU has demonstrated its ability to speed training. Because it may substantially accelerate SGD convergence, the ReLU function, gradually became a preferred Standard to be used option. Furthermore, the function's performance is unaffected by vanishing or expanding gradients, and the function uses reasonable operations rather than expensive exponentials. The ReLU function, on the other hand, has substantial downsides, such as eliminating information that is bad and not working well across datasets and architectures in general. If the output is less than zero, the ReLU activation function invariably returns 0; alternatively, it returns the very same value as the input[18].

3.4 Optimization Algorithms

3.4.1 Stochastic Gradient Descent (SGD)

One of the primary optimization algorithms being used in our Network and most useful for CNN's. There are also some examples of how to calculate a gradient of such parameters in relation to the loss function [2].

$$\theta_{t+1} = \theta_t - \lambda \cdot \nabla\theta_t L\left(f_{\theta_t}\left(x_i\right), y_i\right)$$

It has been claimed that the algorithm's stochastic nature allows it to optimize different loss functions and reduce poor minima. The parameters are randomly initialized, despite the fact that this algorithm achieves good local minima. Furthermore, many local minima are nearly as precise as global minima.

3.4.2 Adam

Adam is a popular optimizer strategy since it gives favorable results in a shorter amount of time. It's been utilized to modify the network weights in training data in an iterative manner. Adam has various advantages, including ease of implementation, great efficiency, and low memory requirements[4]. The mathematical expression of Equations for the Adam optimizer is as follows.

$$m_t = \beta_1 m_{t-1} + \left(1 - \beta_1\right) g_t$$
$$v_t = \beta_2 v_{t-1} + \left(1 - \beta_2\right) g_t^2$$

To converge fast, this algorithm makes use of the advantages of both RMSProp and AdaGrad[21].

3.5 Loss Functions

3.5.1 Mean Square Error Loss

MSE is a multi class loss/cost function that is used while training neural networks[1].

$$\text{Loss}(q, r) = \frac{1}{m} \sum_i \left|q_i - r_i\right|^2$$

In the above equations q is an m-prediction vector, and r is a binary vector with 0 and 1 in based on class dimension.

3.5.2 Cross Entropy Loss

This function tends to outperform MSE[1].

$$\text{Loss}(m, n) = -\sum_i n_i * \log\left(\frac{\exp\left(m_i\right)}{\sum_j \exp\left(m_j\right)}\right)$$

Here, n is just a binary vector consisting of 0 except for a 1 in the associated class dimension, and m would be a vector of n predictions in the preceding equation. The Cross Entropy is desirable because, while MSE loss will eventually inhibit learning, the Cross Entropy will not.

3.6 Workplan

The primary objective of our research is the early detection of colon cancer using Histopathological Images. Proper data had to be collected which would help us get the required result. We collected our dataset LC2500 from Borkowski et al.[23] which contained annotated Whole Slide images on Colon adenocarcinomas and Benign Colon Cells. The data was split into Training, Validation and Test sets for 80:10:10 ratio respectively. Data augmentation was done on the training images to help increase the accuracy of the training model and reduce overfitting. Feature extraction was carried out by first feeding the images through a convolution layer, then maxpooling layer.

This method was performed numerous times depending on the designs of the architecture being used. The resulting data set was flattened and fed into a fully connected neural network. Final Output gave us a Training, Validation and Test score for the performance of the model architecture which came to 96.3%, 95.0% and 94.5% respectively. After which the model are undergone Explainable AI algorithms to get explainable results. The explanations are then analyzed, accuracies are compared and optimization to the models are done accordingly.

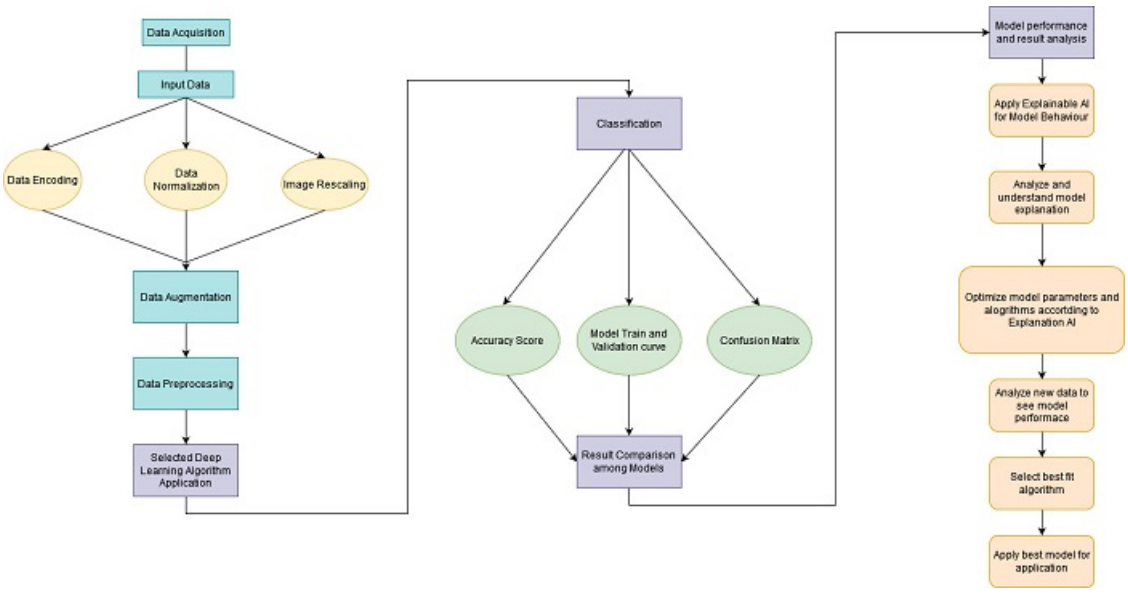


Figure 3.1: Work plan progress for classification of Colon Cancer tissue images using CNN models and explainable AI

Chapter 4

Implementation of Existing Model Architectures

4.1 Baseline Model

LC25000 dataset is used to train our model which has an input of 80 × 80 × 3 dimension. The baseline model uses 3 convolutional layers each with a 3x3 filter kernel and the same padding. The first, second and third layers have 32,64 and 128 filters respectively. ReLU activation is applied in each layer to improve the performance of nonlinear processes. Following each convolutional layer comes a Max-Pooling layer, which is used to downscale the output images. The padding was kept at "same" and the pooling size was kept at 2. The most basic MaxPooling2D operation is used by all pooling layers. We also incorporated a flatten layer, used for turning the convolutional layer's output into a 1D tensor which is connected to a dense layer. Finally, for colon cancer classes of benign or adenocarcinoma, we have a Dense layer that treats the input as a basic vector and outputs a vector, with one Dense layer with 256 neurons and the second layer with 2 neurons. The softmax activation algorithm activates the output layer and a dropout layer is kept in between dense layers with a value of 0.2 as parameter as a prevention to overfitting. The function of this layer is to randomly drop neurons from visible and hidden layers in case of overfitting. Dropout layers are placed in the second and third layers in our model.

4.2 VGG-16

VGG-16 is a very simplistic CNN architecture. With 16 input layers and 224 x 224 RGB image is the construct of VGG-16 model[5]. No weighted hyperparameters we used. The traditional model architecture was kept with the output layer containing 3 dense layers, 2 of which are activated with the ReLU Function and the final one being a softmax output function having 2 output classes. The batch size was set to 64. Both the loss and accuracy start to become steady at around 20 epochs.

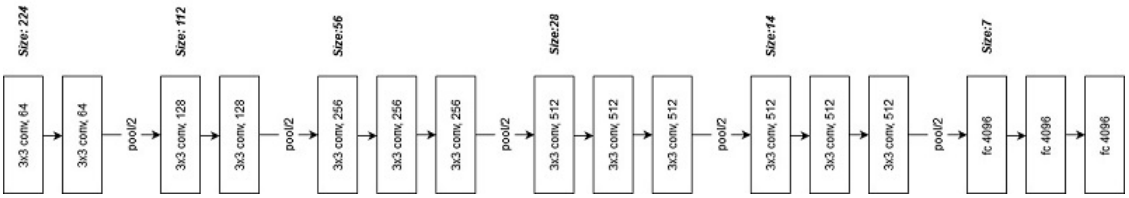


Figure 4.1: VGG-16 Architecture

We achieved an accuracy of around 96.7%. We do however see some sudden spikes in the validation set whilst the training set curve is quite smoothly levelling out indicating there might be some overfitting of the model.

4.3 Resnet50

To make the training period less onerous, the layers have been deliberately altered to learn residual functions with input references rather than unreferenced functions[9]. The required underlying mapping in this case is H(x), The stacked nonlinear layers fit the G(x) = H(x)x mapping, and the layered nonlinear levels fit the G(x) = H(x)x mapping. As a result, the actual mapping is now G(x) + x. The equation G(x) + x has been proposed to reduce the limitation by feeding forward to neural networks building the proposed bypass connections. The skip connections were used to complete the identity mapping and to combine the stacked layer results. The convolutional layers of the architecture typically have 3 3 filters, with a stride value of 2. The network's end has been enhanced with an average global pooling layer, and n-way (category number) fully-connected layer, and softmax. ResNet-50 is a deep convolutional neural network with 50 layers. The ResNet-50 model is made up of five steps in each of the convolution and identity blocks. Each identity block contains three convolution layers, and each convolution block has three convolution layers as well[32].

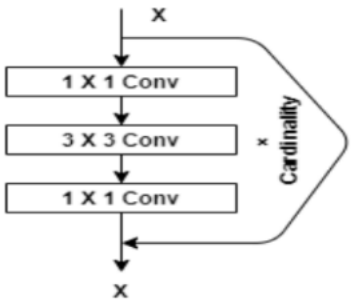


Figure 4.2: Resnet 50 Block Architecture

Our results with ResNet-50 dealt Training accuracy of 98.8% with only 10 epochs.We can see some variance between the validation and training results which would indicate there is a degree of overfitting in the model.

Chapter 5

Proposed Methodology

In existing Deep learning models to detect early stage colon cancer CNN algorithms have been used on histopathological images to train classifiers to identify whether a cell is cancerous or not but there it seems there is always a window of overfitting for the models whereas they fit exceptionally well while training on the dataset but end up making mistakes when generalized images are given for classification.

Our proposed method suggests the inclusion and implementation of explainable AI algorithms with the model, whereas the specific decisions the model took to arrive at its decision can be understood.

5.1 Additive Feature Attribution Methods

To explain a model that is complex and enigmatic we need to use a simple model as a structure to further understand the said complex one. Here,f and g are the variables which refer to the explanation model and the model to also be explained respectively. Local methods are used to explain a prediction *f(x)* with a single input x proposed in LIME [14]. Explanation models use non-complex inputs x', via a mapping function, which are translated to the original inputs $x = h_x(x')$. These methods make an effort to ensure that $g(z') \approx f(h_x(z'))$ while $z' \approx x'$. While applying these it should be kept in mind that $h_x(x') = x$ despite x' having less information than x since h_x is specified based on x which is the current input.

Definition
A linear function of binary variables is used as an explanation model which additive feature attribution methods have:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i \quad .....(i)$$

where $z' \in \{0,1\}M$, M being the number of simplified input features, and $\phi_i \in R$

The explanation models that match the definition stated above relate the factor of ϕ_i for each feature and the cumulative influence of these factors give us an approximate result *f(x)* of the original model. The methods that match the Definition are discussed below.

5.1.1 LIME

Individual model predictions are interpreted using the LIME technique, which involves locally estimating the model around a specific prediction [14]. LIME's local linear explanation model follows equation *(i)* to the letter, making it an additive feature attribution technique. LIME labels x' as simplified inputs and the $x = h_x(x')$ mapping translates binary to decimal vectors of understandable inputs into the input space that was created. Different types of h_x mappings are used for different input spaces. h_x converts a vector of 1s or 0s (existent or otherwise) through into native number of words if the streamlined input is one, or zero if the streamlined input is zero for bag of words text features. When it comes to images, h_x treats them as a collection of super pixels, mapping 1 to keep the original value of the super pixel and 0 to replace it with an average of the surrounding pixels.

LIME minimizes the objective function below in order to find ϕ

$$\xi = \arg\min L(f, g, \pi_{x'}) + \Omega(g).g \in \mathcal{G} \quad .....(ii)$$

Trust of the explanation model $g(z')$ to the original model $f(h_x(z'))$ is effectuated through the loss of L over a set of samples which are weighted by the local kernel $\pi_{x'}$. Moreover, the complexity of g is penalized by Ω. Since g follows equation *(i)* in LIME with L being the squared loss,equation *(ii)* can be solved by a penalized linear regression.
The explanatory model's credibility $g(z')$ in comparison with the original model $f(h_x(z'))$ is effectuated because of the loss of L during the course of a number of samples s which are weighted by the local kernel $\pi_{x'}$. Moreover, the complexity of g is penalized by Ω. Since g follows equation *(i)* in LIME with L being the squared loss,equation *(ii)* a penalized linear regression can be used to solve this problem.

5.1.2 DeepLIFT

DeepLIFT was recently proposed as a deep learning recursive prediction explanation approach [16], [19]. It assigns a value $C_{\Delta x_i \Delta y}$ to each input x_i that is the result of changing the value of the that input to the a reference value instead of its original value. This implies that mapping $x = h_x(x')$ converts binary values into DeepLIFT original inputs, with 1 indicating that an input keeps its initial price and 0 indicating that it keeps its reference value. The level of significance, despite being supplied by the user, is an usual uninformative backdrop worth for the feature.

Summation-to-Delta property of DeepLIFT:

$$\sum_{i=1}^{n} C_{\Delta x_i \Delta o} = \Delta o \quad .....(iii)$$

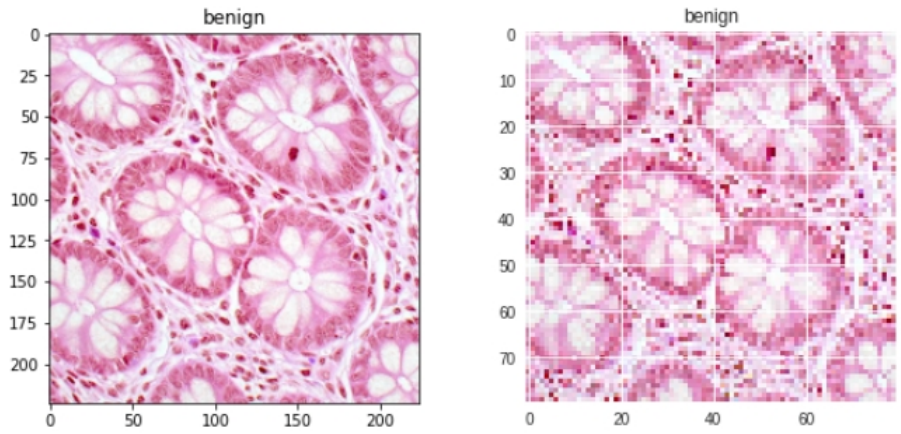
Here, $o = f(x)$ is the model output, r is the reference input with $\Delta o = f(x) - f(r), \Delta x_i = x_i - r_i$ DeepLIFT's explanation model will match equation *(i)* when $\phi_i = C_{\Delta x_i \Delta o}$ and $\phi_0 = f(r)$ and can serve as an another additive feature attribution method.

Chapter 6

Experimental Setup

6.1 Data Preprocessing

For the LC25000 dataset, it consists of both lung and colon cancer images. For our research we only used the colon cancer images. All the images have been kept in RGB format. The original images are of sizes of 1024x768, which were then transformed into squares of 768x768 pixels. For further processing the images were converted to 80 x 80 with 3 color channel space for RGB. When the images are sent to a deeper layer at their current size, image features would be lost, there will be a vanishing gradient problem, and errors will be introduced further into cost function. As a result, we attempted to reduce the size of our dataset to the smallest possible size while maintaining the highest level of visual detail [13]. We have used OpenCV library resizing tools to resize our images to our required criterias for processing and return them to an array



(a) Image before resizings (b) Image after resizing

Figure 6.1: Dataset images before and after rescaling and processing

6.2 Data Augmentation

Various deep learning architectures, such as VGGNet, ResNet, and Inception, are used in this research. The training data set has been supplemented with data for all architectures kept the same [25].

Rescale = image size/255
featurewise std normalization = True
Color Normalization = True

Table 6.1: Augmentation Parameters

6.3 Regularization

Deep neural networks have the potential to retain any type of information. Throughout training, the model's accuracy on the training set tends to increase, while its accuracy on the validation and test set tends to decrease. This behavior is referred to as overfitting. In these cases we can say the model has become overconfident with the training set provided to it and fails to generalize to other general samples.
For small datasets such as LC25000, Overfitting is a serious problem, and the first step in removing any inconsistencies is to tweak the model by adding weight decays or some form of bias to each dimension's cost function, which penalizes the parameters.

$$\mathrm{Err}(m, n) = \mathrm{Loss}(m, n) + \textstyle\sum_i \theta_i^2$$

In this equation, is a vector that contains all of the network parameters.

6.4 Dropout

We also included dropout layers to prevent overfitting. In our model a dropout layer is kept in between dense layers with a value of 0.2 as parameter as a prevention to overfitting. Neurons are randomly dropped from visible and hidden layers to avoid overfitting.

6.5 Early Stopping

This strategy primarily looks for signs that a model's performance is deteriorating, such as a lack of improvement in accuracy or an increase in loss. So that the training can be stopped after a particular number of epochs. As a result, it is beneficial to halt a model's training if it begins to overfit or underfit.

6.6 Training/ Validation/ Test Set split

The dataset has taken 10,000 images with 5000 benign and 5000 adenocarcinoma for colon Whole Slide Images. We took a dataset split of 70% training set, 20%

cross validation and 10% test set. We kept our sample the same for the validation set, except we used rescale to normalize the images= image size / 255.

6.7 Evaluation metrics

To test our models performance we used accuracy scores along with confusion matrices . Accuracy is calculated with the action of dividing the total number of true predictions by the total number of categories predicted.

$$\text{Accuracy } = \frac{TP+TN}{TP+TN+FP+FN}$$

TP, TN, FP and FN in the equation denote true positive, true negative, false positive and false negative respectively.

As we have 2 classes, we have created a 2x2 confusion matrix with benign and malignant classes as the labels. The column denotes the predicted classes while the rows are referred as the actual class.

Actual	True Positive	False Positive
	False Negative	True Negative
Label	Predicted Label	

Table 6.2: Confusion Matrix for colon cancer classes

Chapter 7

Result and Evaluation

7.1 Implementation of Deep learning Model

For hyperparameter tuning, we've used a variety of optimizers and feature tuning methodologies. However, the two types of optimizers shown in Table (unspecified), each with a batch size of 64, a weight decay value of 0.0001, and different learning rates for each optimizer, produce well-balanced results, we trained our model for 20 epochs in our Baseline Model, 100 epochs in the VGG-16 model, and 10 epochs in the ResNet50 model. It is observed that all 3 models perform relatively the same

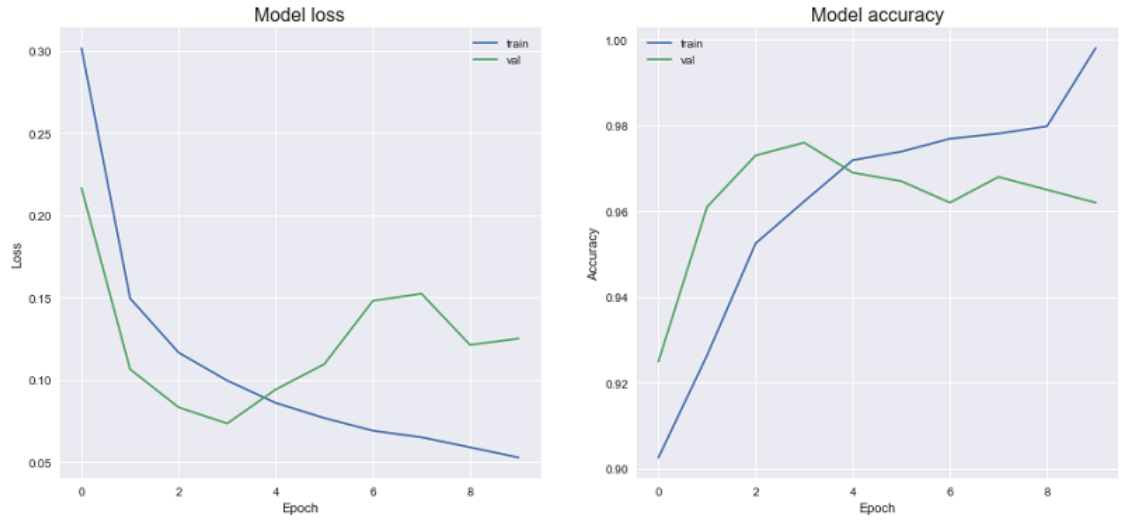
Model Architectures	Batch Size	Learning Rate	Optimizers	Accuracy
ResNet50	64	0.01	SGD	92.1%
		0.001	Adam	96.7%
VGG-16	64	0.01	SGD	95.4%
		0.001	Adam	96.7%
Base Model	64	0.01	SGD	94.5%
	64	0.001	Adam	96.2%

Table 7.1: Deep Learning feature tuning results

for both optimizers with a different of 3-4%. We can also see that SGD tends to be performing a bit poorer than Adam optimizer.

7.2 ResNet50

For ResNet50, the SGD optimizer gave an accuracy of 92.1% with some gap between the convergence of the training set and test set, which would indicate that there is an overfitting problem.



(a) Resnet50 Loss

(b) Resnet50 Accuracy

Figure 7.1: Accuracy and Loss Graph for Resnet50 Model Architecture

Our assumption is that this might be due to the dataset being too small for ResNet50 architecture. For Adam optimizer on the other hand things were quite different. Both validation and Training set converged quite quickly within just 10 epochs and gave an accuracy of 96.7%.

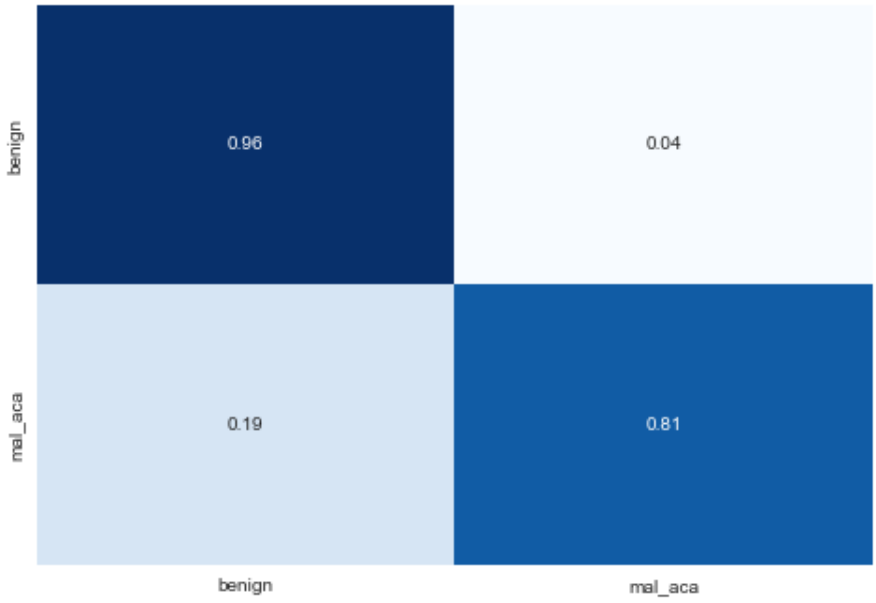
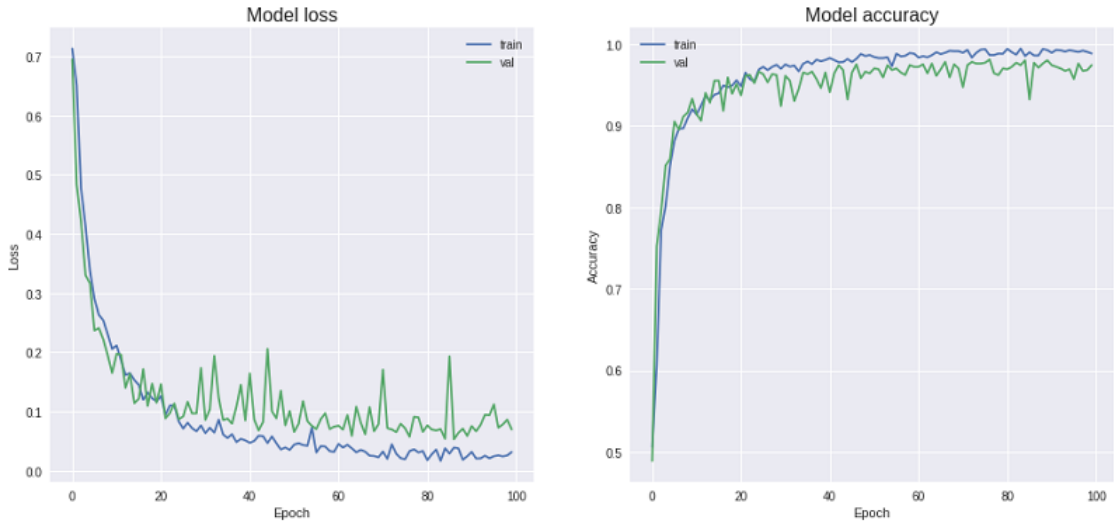


Figure 7.2: Confusion matrix for ResNet 50

7.3 VGG-16

For VGG-16, the SGD optimizer gave an accuracy of 95.4% with both validation and training curves converging relatively well. We do however observe some spikes

in the loss function of the validation curve which also resulted in the accuracy for the validation set to have huge spikes in wrongful classification.



(a) VGG-16 Loss

(b) VGG-16 Accuracy

Figure 7.3: Accuracy and Loss Graph for VGG-16 Model Architecture

This might be anomalies that occurred due to the small size of the dataset. More about wrongful classification will be discussed ahead. With Adam the validation and training set both converged nicely but also had some spikes in the validation set, though not as much as the SGD optimizer. The accuracy for Adam came at 96.7%.

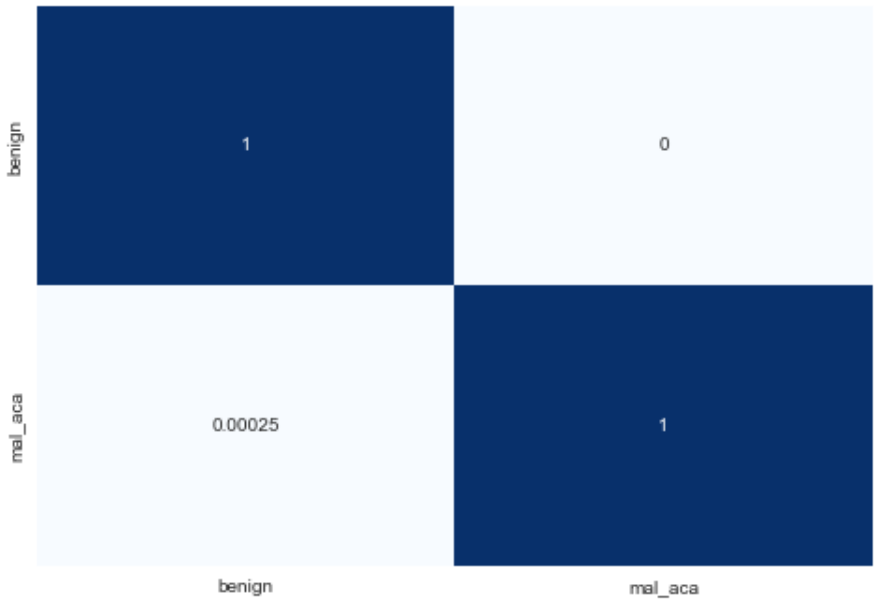
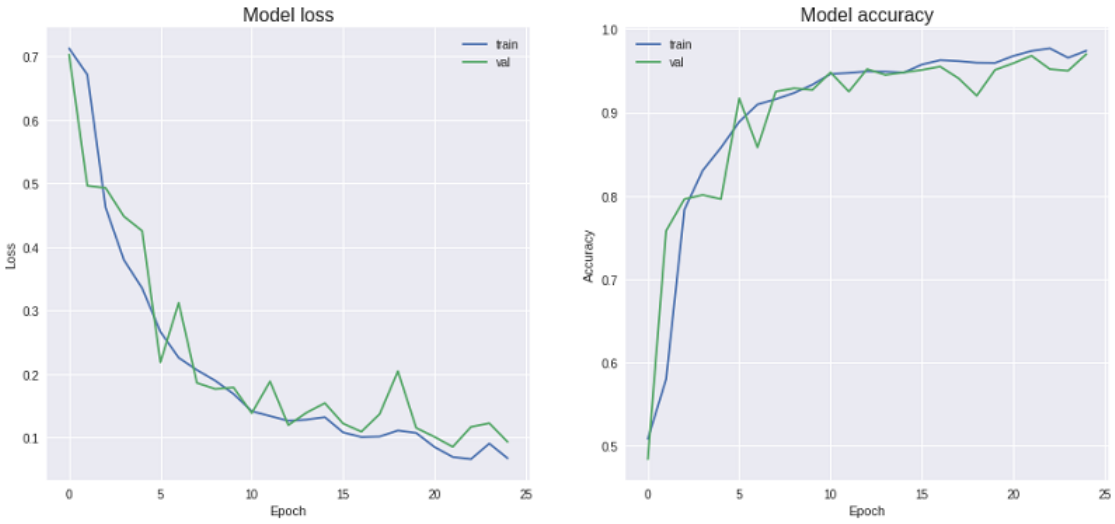


Figure 7.4: Confusion matrix for VGG-16

7.4 Baseline Model

Finally, for our custom Baseline Model, the SGD optimizer gave us an accuracy of 94.5% whilst the Adam optimizer gave an accuracy of 96.2%.



(a) Baseline loss (b) Baseline Accuracy

Figure 7.5: Accuracy and Loss Graph for Baseline Model Architecture

Even though the accuracy of the model is lower than that of the other pretrained models, the validation and training curve converged at a much smoother rate than the other models, even though this too had some random spikes in the validation set, which would indicate that the model was less prone to overfitting and even if gave lower accuracy score, gave us more authentic results.

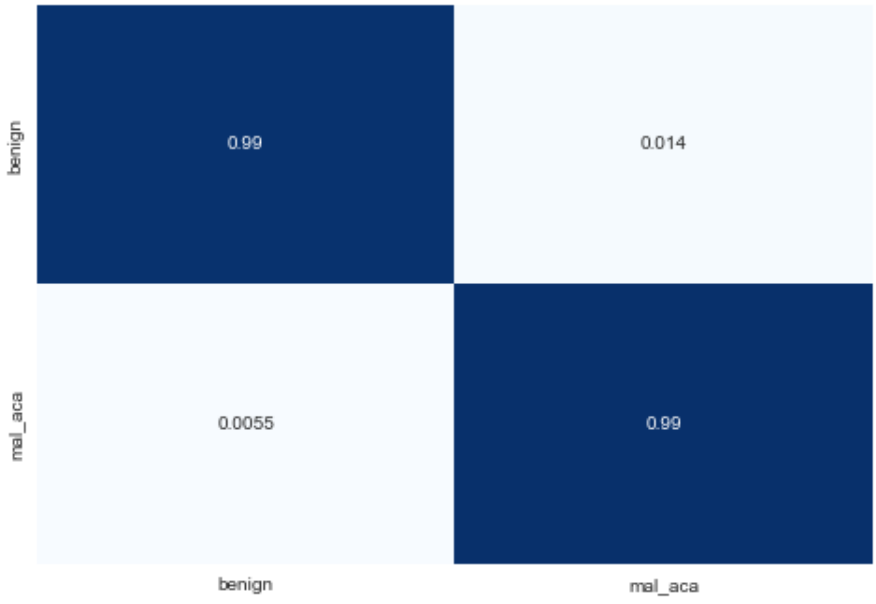


Figure 7.6: Confusion matrix for custom Baseline Model

Overall we understood that our baseline model performed quite admirably despite being a lot less complicated than the other models. Though it must be taken into account that the dataset that we have used is quite small, within only 10,000 samples being taken. The other pre-trained architectures are made for much larger datasets of possibly hundreds of thousands of samples, which would explain their overfitting issues as well as the sudden spikes in the validation set.

7.5 Wrongly classified images

With further investigation we extracted some random false negative images from our test set, classified wrongly by our baseline model.

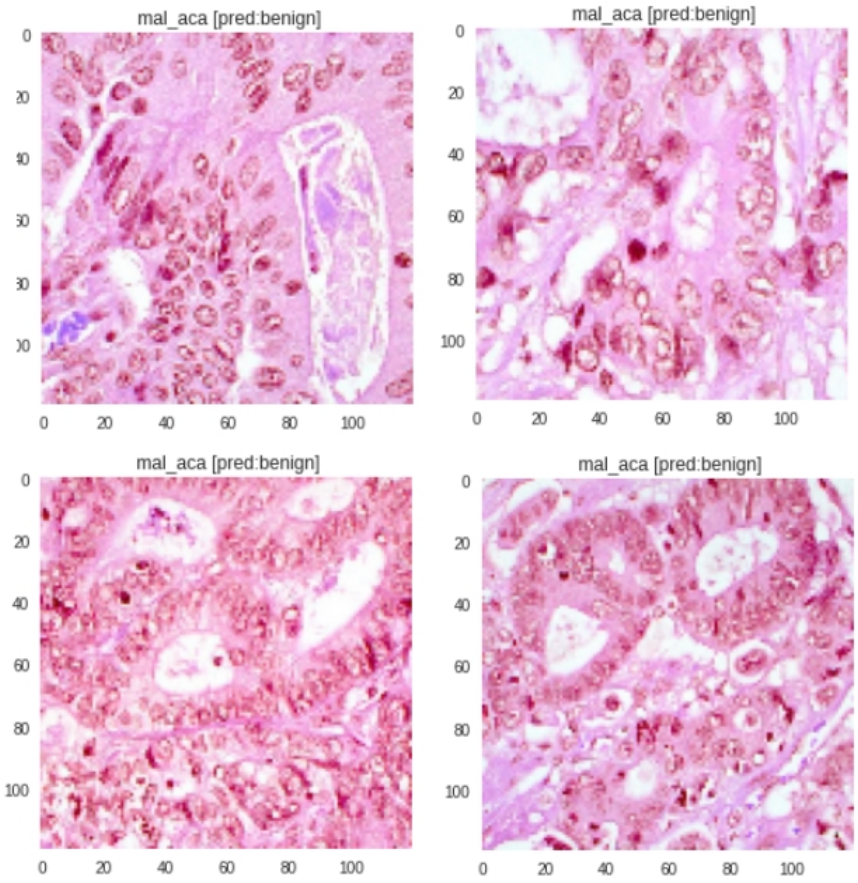
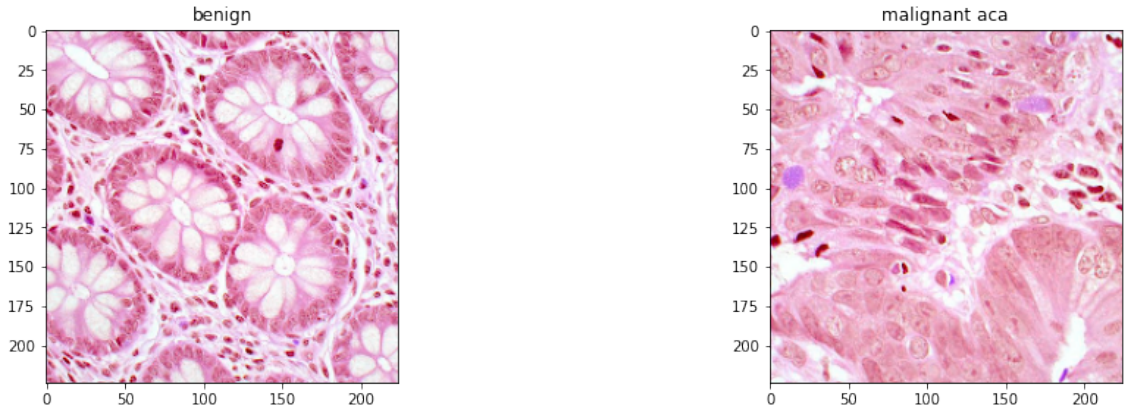


Figure 7.7: Wrongly classified malignant cancerous cells

The above figures are images of malignant cancerous cells afflicted with adenocarcinoma. Our classifier has classified them all as benign tumours, which is a false negative. In order to understand why our classifier made the mistake and what optimizations can be done to make the classifier more accurate in identifying these cells, we took a different approach to understanding the algorithm by using Explainable AI modules.

7.6 Implementation of Explainable AI

As we saw, all our models were prone to some sort of overfitting as well as wrongful classifications of samples. In order to understand why we implemented explainable AI in our models to understand how they took their decisions and how they classified each sample.



(a) Benign tumor cells

(b) Malignant Cancer Cells

Figure 7.8: Benign and Malignant adenocarcinoma cells

On the left in the above image, we see a benign tumor cell with well-rounded cell structures, and on the right, we find a malignant tumor cell with adenocarcinoma with cell structures obliterated.

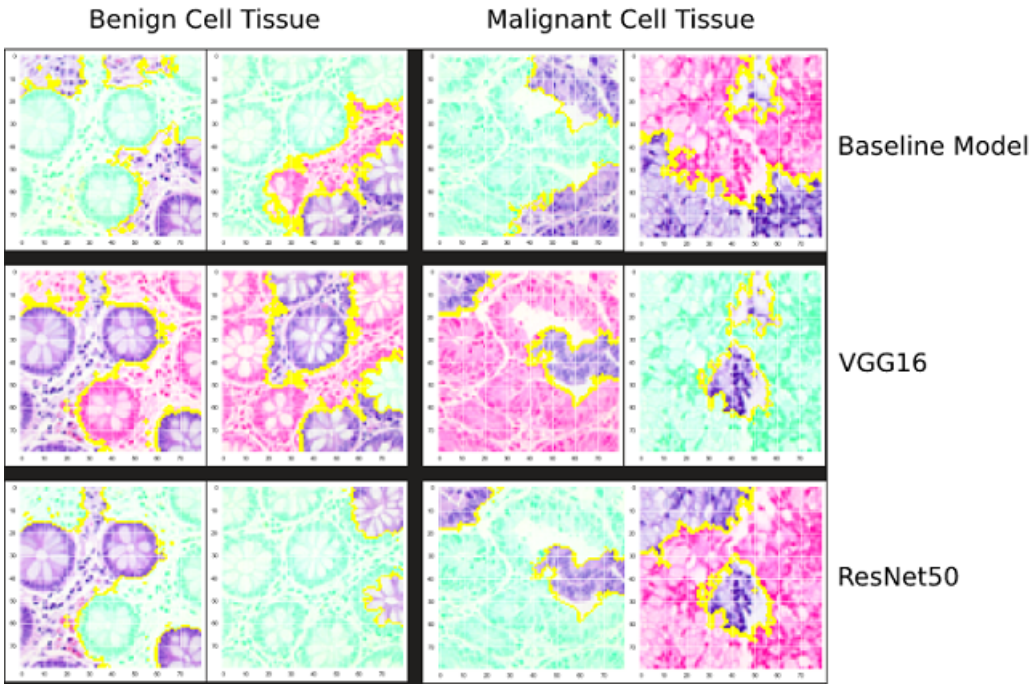
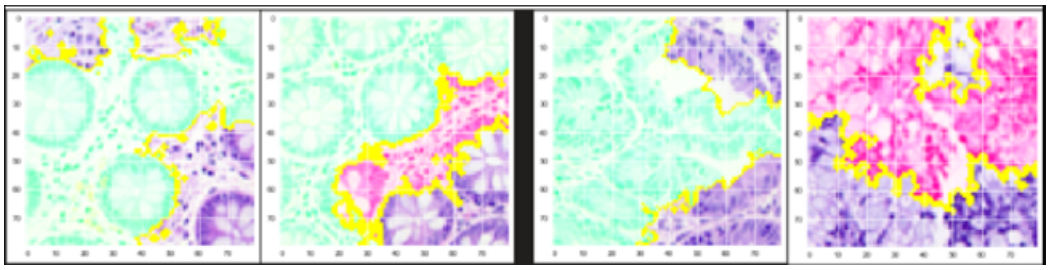


Figure 7.9: Comparison between Benign and Malignant cell tissue classified by all three classifiers with Explainable AI library

The above figure shows us the method of classifications made by each model archi-

tecture.

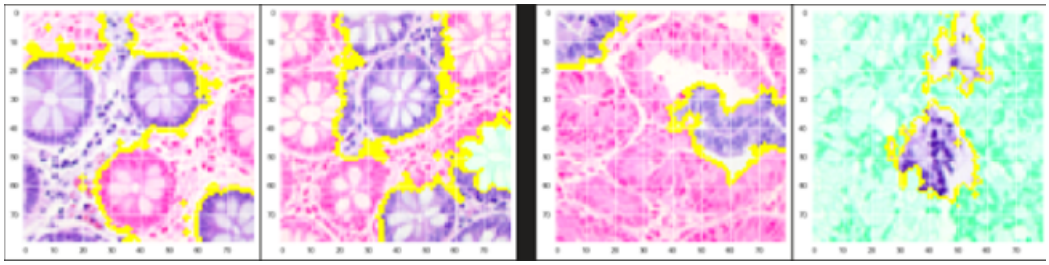
The color classifications are with the light green classified colors being predicted Malignant Cancer cells, the Crimson classified color being predicted as Benign Tumour cells and all other parts of the cell are undetected.



(a) Benign tumor cells (b) Malignant Cancer Cells

Figure 7.10: Visual explanations for Baseline model Architecture

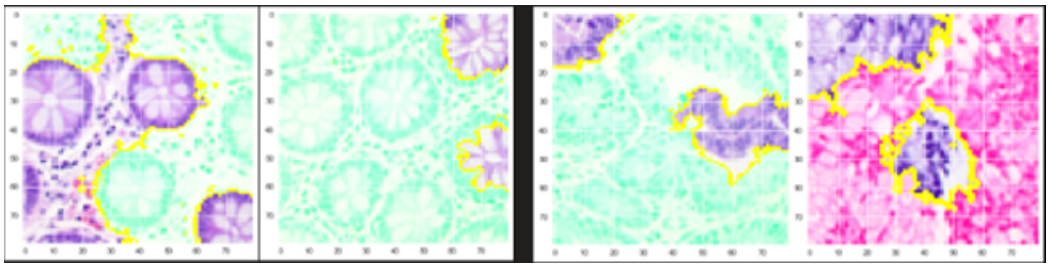
In the Baseline model we can see that for the Benign Tumour cells, the classifier classified nearly 70% of the cells as cancerous even though it is benign and for Malignant cancer cells, the model classified the image on the left quite accurately but the image on the right classified the image as completely benign which is entirely wrong. This tells us that the sudden spikes we see in our validation curve are due to some samples being extremely overfitted to the dataset.



(a) Benign tumor cells (b) Malignant Cancer Cells

Figure 7.11: Visual explanations for VGG-16 model Architecture

In the VGG-16 model we see that the model has classified the benign tumour cells perfectly with no anomalies being detected. This proves that whilst the model did have overfitting issues, its higher number of layers and more sophisticated architecture did allow it to fit better with the dataset. On the Malignant cells classification we see that it has accurately classified one of the images as being cancerous whilst the other image has been classified completely false. This explanation through visual data will allow us to make the model more precise that it already is with some slight optimizations.



(a) Benign tumor cells (b) Malignant Cancer Cells

Figure 7.12: Visual explanations for ResNet50 model Architecture

In the ResNet models the classifications seem to be more like the baseline model. For ResNet50 we can see that for the Benign Tumour cells, the classifier classified nearly 70% of the cells as cancerous even though it is benign and for Malignant cancer cells, the model classified the image on the left quite accurately but the image on the right classified the image as completely benign which is entirely wrong. This tells us an interesting fact that whilst ResNet did perform very well with the training set, of all the models, it has the highest overfitting issues, possibly because of the small dataset. The AI explanations provided by the images tells us that even though we ran ResNet for only 10 epochs, it had still overfitted to quite an unacceptable degree. This allows us to assume that in case of ResNet a much much larger dataset is required for better accuracy rather than just fine optimizations.

7.7 Comparison of Accuracies for applied model architectures

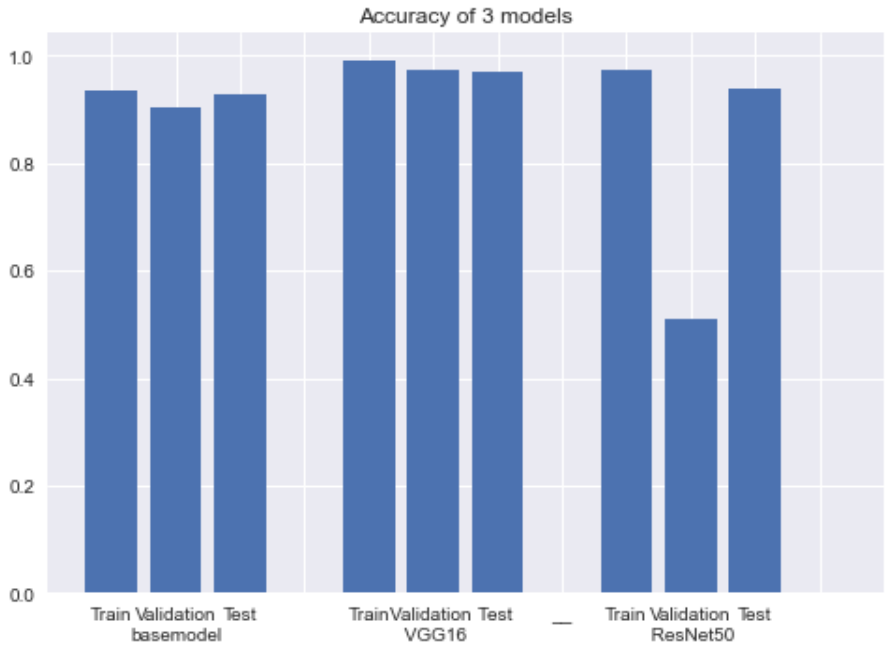


Figure 7.13: Comparison graph of accuracy for all tested models

Figure 7.14: Comparison graph of accuracy for different optimizers

Comparing the results of all 3 architectures we can say that our base model is performing quite well with an average of above 94% accuracy with some slight differences between the validation and training set. ResNet50 model as can be observed is overfitting quite heavily on the training set and performing very poorly on the validation set. This lets us know that perhaps ResNet50 will require more than just optimization but rather a larger dataset in order to be more reliable. So far VGG-16 is perhaps the best performing model with consistent behaviour among all 3 sets.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

Early stage detection of colon cancer is a vital research field for not only the advancement of science but also to save lives. In this paper we have implemented a Deep Learning algorithm on Colon Cancer Dataset of Histopathological Images, LC25000 Dataset. We ran on the dataset, both pre trained and custom made baseline CNN Deep Learning models to identify and categorise between Benign and Malignant Cancer cells. Our results show us that all deep learning models perform quite well in classifying the tumour cells with an above average accuracy of more than 94% on the test set.

Even still we came across some overfitting in all the models we implemented. In order to get a better understanding and to open a path for further optimization we took a different approach, with Explainable AI algorithms, where we were able to see the explanations provided by the AI, in visual image format, for the specific decisions and classifications it had done. Through these explanations we were able to understand the pros and cons of each deep learning model on a more crucial level.

8.2 Future Work

Explainable observations allowed us to understand that our baseline model is far from ideal and requires more fine tuning even if it performed quite admirably. Resnet50 architecture performed relatively the same or worse than our baseline model with high levels of overfitting. The explanations helped us understand that pre trained models such as ResNet 50 and VGG-16 need a larger dataset to perform better along with more computational power as this is the the biggest impediment to this project is a lack of resources that can provide us with computational capability, namely processing power constraints (GPU requirements) and a lack of credible research and datasets.

Despite that we were able to make a new classifier optimization approach towards the detection of colon cancer using histopathological images and Deep Learning. There are further issues to be addressed for the future. In our further research we would like to use our AI provided explanations to further optimizations and enhance our pretrained model as well as modify and add more features to our baseline model and

run it on a much larger and more reliable dataset in order to avoid overfitting issues and attain better performance. Also, in the future, we will use grid search-like methods to discover the best-fitting combination of hyperparameters for optimal accuracy so that our model can identify any quality generalized image of colonic cancer and correctly classify them.

Bibliography

[1] D. M. Kline and V. L. Berardi, "Revisiting squared-error and cross-entropy functions for training neural network classifiers," *Neural Computing & Applications*, vol. 14, no. 4, pp. 310–318, 2005.

[2] L. Bottou, "Stochastic gradient descent tricks," in *Neural networks: Tricks of the trade*, Springer, 2012, pp. 421–436.

[3] S. A. Hussain and R. Sullivan, "Cancer control in bangladesh," *Japanese journal of clinical oncology*, vol. 43, no. 12, pp. 1159–1169, 2013.

[4] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free?-weakly-supervised learning with convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 685–694.

[7] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[8] P. Favoriti, G. Carbone, M. Greco, F. Pirozzi, R. E. M. Pirozzi, and F. Corcione, "Worldwide burden of colorectal cancer: A review," *Updates in surgery*, vol. 68, no. 1, pp. 7–11, 2016.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[11] E. P. Ijjina and C. K. Mohan, "Hybrid deep neural network model for human action recognition," *Applied soft computing*, vol. 46, pp. 936–952, 2016.

[12] A. Janowczyk and A. Madabhushi, "Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases," *Journal of pathology informatics*, vol. 7, 2016.

[13] K. K. Pal and K. Sudeep, "Preprocessing for image classification by convolutional neural networks," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, IEEE, 2016, pp. 1778–1781.

[14] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[15] S. Sarraf and G. Tofighi, "Deep learning-based pipeline to recognize alzheimer's disease using fmri data," in *2016 future technologies conference (FTC)*, IEEE, 2016, pp. 816–820.

[16] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," *arXiv preprint arXiv:1605.01713*, 2016.

[17] K. Sirinukunwattana, S. E. A. Raza, Y.-W. Tsang, D. R. Snead, I. A. Cree, and N. M. Rajpoot, "Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1196–1206, 2016.

[18] H. Ide and T. Kurita, "Improvement of learning for cnn with relu activation by sparse regularization," in *2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2017, pp. 2684–2691.

[19] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International Conference on Machine Learning*, PMLR, 2017, pp. 3145–3153.

[20] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, "Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: a cancer journal for clinicians*, vol. 68, no. 6, pp. 394–424, 2018.

[21] S. De, A. Mukherjee, and E. Ullah, "Convergence guarantees for rmsprop and adam in non-convex optimization and an empirical comparison to nesterov acceleration," *arXiv preprint arXiv:1807.06766*, 2018.

[22] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," *arXiv preprint arXiv:1811.03378*, 2018.

[23] A. A. Borkowski, M. M. Bui, L. B. Thomas, C. P. Wilson, L. A. DeLand, and S. M. Mastorides, "Lung and colon cancer histopathological image dataset (lc25000)," *arXiv preprint arXiv:1912.12142*, 2019.

[24] M. Shapcott, K. J. Hewitt, and N. Rajpoot, "Deep learning with sampling in colon cancer histology," *Frontiers in bioengineering and biotechnology*, vol. 7, p. 52, 2019.

[25] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[26] S. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, "Impact of fully connected layers on performance of convolutional neural networks for image classification," *Neurocomputing*, vol. 378, pp. 112–119, 2020.

[27] O. Iizuka, F. Kanavati, K. Kato, M. Rambeau, K. Arihiro, and M. Tsuneki, "Deep learning models for histopathological classification of gastric and colonic epithelial tumours," *Scientific Reports*, vol. 10, no. 1, pp. 1–11, 2020.

[28] S. Mangal, A. Chaurasia, and A. Khajanchi, "Convolution neural networks for diagnosing colon and lung cancer histopathological images," *arXiv preprint arXiv:2009.03878*, 2020.

[29] P. Sabol, P. Sinčák, P. Hartono, *et al.*, "Explainable classifier for improving the accountability in decision-making for colorectal cancer diagnosis from histopathological images," *Journal of Biomedical Informatics*, vol. 109, p. 103 523, 2020.

[30] R. L. Siegel, K. D. Miller, A. Goding Sauer, *et al.*, "Colorectal cancer statistics, 2020," *CA: a cancer journal for clinicians*, vol. 70, no. 3, pp. 145–164, 2020.

[31] L. Xu, B. Walker, P.-I. Liang, *et al.*, "Colorectal cancer detection based on deep learning," *Journal of Pathology Informatics*, vol. 11, 2020.

[32] C. Giuseppe, "A resnet-50-based convolutional neural network model for language id identification from speech recordings," in *Proceedings of the Third Workshop on Computational Typology and Multilingual NLP*, 2021, pp. 136–144.