A Decentralized Learning-Based Approach To Classify Colorectal Cancer Using Deep Learning Leveraging XAI

by

Kazi Ehsanul Mubin
18101391
Noshin Tabassum Arthi
18101100
Junayed Rahman
18101095
G. M. Rafi
18101465
Tahsina Tanzim Sheja
18101504

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
May 2022

© 2022. Brac University
All rights reserved.

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:

Kazi Ehsanul Mubin
18101391

Noshin Tabassum Arthi
18101100

Junayed Rahman
18101095

G. M. Rafi
18101465

Tahsina Tanzim Sheja
18101504

Approval

The thesis titled "A Decentralized Learning-Based Approach to Classify Colorectal Cancer using Deep Learning Leveraging XAI" submitted by:

1. Kazi Ehsanul Mubin(18101391)
2. Noshin Tabassum Arthi (18101100)
3. Junayed Rahman(18101095)
4. G. M. Rafi(18101465)
5. Tahsina Tanzim Sheja(18101504)

As of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering in May, 2022.

Examining Committee:

Supervisor:
(Member)

Md. Ashraful Alam, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

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

Thesis Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

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

Abstract

Convolutional Neural Networks (CNN)-based automated approaches are vastly utilised to anticipate and diagnose cancer, saving time and reducing mistakes. Deep Learning (DL) CNN methods use a variety of probabilistic and statistical methodologies to make computers understand and identify patterns in datasets based on previous experiences. We proposed an efficient federated learning based model to classify histopathological images for detecting colorectal cancer while providing high prediction accuracy and maintaining data privacy. Federated learning solves the problem of retaining privacy while utilizing vast and heterogeneous private datasets collected from numerous healthcare facilities. As the amount of patient data obtained for the process of machine learning is significantly responsible for the success of enhancing the accuracy of the system, the experiment was performed on a large dataset including cancerous and non-cancerous colorectal tissue images. FL can also mitigate costs resulting from traditional, centralized machine learning approaches. We have also applied the XAI method, a model-agnostic approach to acquire an explicit demonstration of the applied machine learning models. With XAI, we can visualize the super pixels of our colorectal tissue images through accepting and rejecting features. While applying various CNN models such as VGG16 & 19, InceptionV3, ResNet50, ResNeXt50, and comparing their precision, ResNeXt50 was established with the highest accuracy of 99.53%. Therefore, we have applied ResNeXt50 on FL that brings forth the accuracy of 96.045% and F1 Score is 0.96.

Keywords: Federated Learning, XAI, Deep Learning, Colorectal Cancer, Convolutional Neural Network, Image Classification, ResNeXt50

This research is dedicated to the memory of late A.M.M Bahauddin, an academic, a teacher and a father of one of our closest friends who succumbed to his unfortunate death on December 22, 2021 due to Colon Cancer. He was a lifelong educator and the principal of Feni's Fulgazi Government College. He was respected, loved, honored and admired by everyone he knew. It was his memory that inspired us to conduct this research.

In Memory of:

A.M.M Bahauddin
(1959 - 2021)
Retired Principal
Fulgazi Government College
Feni, Bangladesh

Contents

List of Figures

List of Tables

Acknowledgement

We would like to express our heartfelt gratitude to our supervisor, Dr. Md. Ashraful Alam, and co-supervisor, Md Tanzim Reza, for their unwavering support during our study and thesis preparation. Their prompt advice, careful inspection, and sensible approach have greatly aided us in completing our research. Besides, all of our authors hardwork, insightful discussion, and energetic effort has made the thesis expedition feasible. We would also like to thank the NCT Biobank (National Center for Tumor Diseases, Heidelberg, Germany) and the UMM pathology archive (University Medical Center Mannheim, Mannheim, Germany) for the images and tissue samples. Last but not the least, a big thanks to Marc Macenko for the image color normalization process.

Chapter 1

Introduction

1.1 Background

Colorectal cancer being one of the most common malignant neoplasms, in terms of cancer mortality, it ranked from 2nd to 4th in the globe, based on the region, form of disease, and gender. It also indicates an upward trajectory in terms of morbidity and mortality. Numerous causes, including biological and epigenetic ones, may have a part in the growth of such conditions. Depending on where the coloreactal cancer starts, it can be entitled as either Colon cancer or Rectal Cancer. In our work, we have shaded more light on the colon cancer which is defined as the abnormal proliferation of tissue also known as polyp that is adjacent to the intramural area of the colon. The closest part to rectum and anus is known as the large intestine that holds the most common area for cancer which is called pelvic colon or sigmoid colon. It is the final part of the digestive system which absorbs water and salts from excreta. Abnormal growth of tissue is often classified as a tumor, but when it has the potential to spread and cause harm, it is defined as cancer in Oncology.

Colon cancer is conventionally detected by using certain radiological examinations such as Colonography that can procure an interior view of colon and these images are examined by the radiologists to find polyp-like structures using computer tools. Identifying polyps in a primary stage is pivotal for survival as they might transform into colorectal cancer at a delayed stage. The overall five-year survival rate of colon cancer is around 68% in a specific stage. It was estimated by the American Cancer Society (ACS) in 2014 that 136,830 people were diagnosed with colorectal cancer and 50,310 among them died [1].

An invasive polyp detection procedure, Colonoscopy is quite time demanding and expensive that requires high quality bowel preparation and air insufflation during examination. The results brought by these examinations are constructed by the finding of the radiologists and endoscopists and can be less accurate which may lead to a missed detection. A certain judgment of the operator not only determines the risk of the patient getting exposed to cancer but also decides the stage of cancer where the patient should get through treatment methodology such as chemotherapy or clinical surgery. Hence, it is vital for the operators to have precise knowledge and expertise to be capable of declaring a patient's condition properly. In the early stages of colon cancer, features such as micrometastasis and polyps are often ignored or misidentified because of their diminutive and sessile appearance. Missed polyps often result in a late diagnosis of Colorectal cancer, lowering the survival chances to 10% under the worst scenarios [2]. This is why this technique of

cancer diagnostics is inefficient and inaccurate.

In recent decades, the use of Computer Aided Diagnostics(CAD) has brought a major ameriolation in cancer detection techniques which is fabricated by automated data curation and annotation of video data. Modern classification and diagnosis of colon cancer is a result of thorough maneuvering of machine learning and deep learning algorithms. Deep learning is a subsector of Artificial Intelligence which works by extracting useful patterns and features from specific datasets to determine an accurate result. Steadily, it has become the most extensively used computational strategy in the field of machine learning, delivering exceptional results on a variety of complex cognitive tasks through it's universal learning approach with scalability and robustness. CNN is the most utilized class of Deep Neural Networks that has an eminent contribution on the detection and classification of colon cancer at different levels. Being a layered artificial neural network, CNN can detect patterns in pixelated images. We conducted our classification task with the decentralized Federated Learning(FL) synced with appropriate CNN models. FL is an adaptive learning paradigm that allows several edge systems to work together to train a global model without relying on a single device's dataset. When an individual company or client lacks the necessary or valid data, FL can help improve accuracy of the model. It's also a viable option when data consolidation is unwanted or impossible owing to privacy or regulative issues.

1.2 Research Problem

Since colon cancer is becoming more common, newer categorization techniques using various algorithms are being introduced to us on a regular basis. It is challenging to accelerate algorithm enhancement in this province due to a lack of available data indices. Furthermore, systematizing medical data repositories is difficult since it necessitates a high level of precision and accuracy in the field [3]. As its phenomena is primarily recognized via shape feature extraction, segmenting colon from noise in the colonography image and eliminating structures that look identical from the genuine polyp is rather complex. These issues can be resolved with extensive pre-processing and color normalization of our training dataset. Macenko's method [4] is an already existing color normalization process for quantitative analysis of histopathological images. We are using a pre-existing dataset that has already been color-normalized using Macenko's method. This initiative should be beneficial to reduce noises in our dataset, resulting in a more efficient system with accurate predictions for our model.

Several studies, including [5], have demonstrated that their models only apply to cases of colon cancer but not in the cases of rectal cancer. It means the use cases for their system have very limited flexibility. Considering this, we will be using a dataset that contains both rectal and colon cancer, known as colorectal cancer tissue images. Moreover, the resulting model is often unable to explain itself and its extracted features for classification. Due to the sheer differences in result patterns, researchers cannot utilize this precise system for other situations such as rectal cancer. Additionally, Polyp detection from photographs is difficult due to large differences in polyp size, color, texture, and other factors [6]. As a result, the system may overlook polyp patches. Besides, microscopic polyps are more difficult to detect than larger ones. Flat or tiny polyps might cause it to be missed up to 22% of the time. The diameter of a little polyp can range from less than a fraction of an inch to several inches. A possible solution to overcome these issues is the implementation of XAI in our model, so that we can easily identify the accepting features of our dataset and training model. As XAl marks the accepting and rejecting features using colored super pixels for classification, we would be able to verify the polyp and micropolyp density using this method more precisely and assume the approximate amount of polyp present in a tissue.

Another study, Angermann et.al. suggested using video sequence analysis to train the system [7]. Although video sequences are not applicable to our approach, the research demonstrates an issue which is that the lack of public data is detrimental and presents barriers to obtaining crucial findings. One reason might be that medical data repositories are heavily regulated. Motivated by this, we are planning to use a federated learning model, which is primarily focused on the privacy concerns regarding medically induced datasets. This model has potential to generate good results although not directly accessible by researchers. Similarly, in another research, author Bae et.al. mentioned that they had a limited number of polyp samples considering there are many different types of existing polyps [8]. Hence there is an imbalanced training and the dataset outcome is biased toward the class label. In the machine learning community this is known as "Class Imbalance Problem". For our FL model we are considering using comparatively large scale clients to avoid this issue.

Overfitting is a typical supervised neural technique flaw where a system needs to integrate all of the datasets and ends up storing the patterns in the data as well as chaos and unpredictable oscillations. An overfitting issue is indicated by a high standard deviation in prediction accuracy. Overfitting can be caused by the model's training phase or structural intricacy. On the other side, underfitting happens due to high bias in data. From recent studies we've noticed that overfitting has been a fundamental issue in many cases. However, to overcome these issues we've used a large dataset and some measurement techniques such as, early stopping, dropout layer etc. early stopping strategy seeks to suspend the model's training before remembering data clutter and unpredictable oscillations. Moreover, In a CNN architecture, large weights indicate a heavy dense network. Preventing prediction error by dropping out network nodes based on probabilities is an easy and constructive strategy. To minimize the model's intricacy, different levels of productions are arbitrarily disregarded during normalization.

A vast database containing the entire range of conceivable anatomical features, diseases, and input feature formats is required to train an AI-based cancer classifier [9]. Because healthcare information is very confidential and data like this is hard to procure. Even if information confidentiality could circumvent such restrictions, it is now widely accepted that deleting schemas such as a patient's personal information is frequently insufficient to protect security. Another reason for the lack of comprehensive information sharing in medicine is that gathering, and having a strong data set requires much time and resources. As a result, such datasets may have important commercial value, making them less likely to be freely shared. Instead, data providers frequently maintain fine-grained control of the information they acquire. To resolve this issue we've applied FL in our research to solve the challenges of data management and security by collectively learning models rather than transmitting information. FL allows for collaborative discoveries, such as in the form of a consensus mechanism, without transferring patient data outside of the institutions' borders. Therefore, each collaborating institution's ML process occurs locally, with only model features being transmitted.

XAI is a collection of strategies that allow researchers to understand and trust advanced techniques, findings and output. An AI model's projected impact and potential flaws are described using XAI. In AI powered strategic planning, it helps describe model correctness, impartiality, clarity, and results. The goal of XAI is to help people understand how machine learning and AI work and how they make decisions. The term "black box" refers to concepts that are so sophisticated that they have been not comprehensible by individuals. To overcome this problem we've come up with the application of XAI in our model, it will recognize the biases in the actions. XAI helps to detect and understand unfairness concerns so that it can be removed. By using this feature medical image analysis can be advanced in future and utilization of image processing techniques will be progressive.

All of the research mentioned above employed CNN models to identify polyps for cancer detection, although it is a really quick method, but this design does not explain why cancer occurs. Thus, XAI is highly effective in this regard. Along with identifying the tissue, it also analyzes why it exists. Moreover, due to confidentiality constraints that surround medical institutes around the world, image data and photos are sparse. As a result, the incorporation of FL models would assure both data protection and CNN model efficacy.

1.3 Research Objective

This research is focused on discussing a decentralized model of a deep learning based CNN framework for detection and classification of Colorectal Cancer using histopathological images collected from publicly available medical data repositories. Precious and timely detection of precancerous growth utilizing machine-driven technologies would assist the patient in receiving suitable treatment in a timely manner, as most cancers are curable only if caught early. Also the scarcity of relative data is taken into consideration in our research, resulting in the implementation of a scalable federated learning model. This decentralized model would ensure the privacy ethics that are mainstream in modern day medical institutions. This research also proposes effective usage of Explainable Artificial Intelligence(XAI) to describe the model that will be followed. The followings are the goal of our paper:

- To understand deep learning and the CNN model in a better way.
- To understand the feasibility of using CNN model in Colorectal Cancer classification over a large dataset.
- To classify colorectal cancer affected images from multi-class tissues.
- To try to attain the best possible results by implementing different CNN models.
- To compare different CNN models and their performance over a selected dataset.
- To implement a scalable federated learning prototype model over our selected models.
- To implement XAI for visualizing the features found in this multiclass classification.
- To further improve and add features to the model in future.
- The system should have consistent performance and enhanced patient variation robustness, as well as the capability to implement credible outputs.

Chapter 2

Literature Review

AI based research, especially improvements in ML and DL, has resulted in game-changing advancements in radiography, histology, genomes, and other domains. Contemporary deep learning models have millions of parameters that must be trained from reasonably large selected data sets to attain medical reliability whilst maintaining security, fair, and equitable, along with generalizing effectively to the changing data. Deep learning, particularly Convolutional Neural Networks, and its application to medical image processing have been thoroughly investigated in recent years. For example, by using the Kvasir-SEG dataset, detecting and sectionalizing polyps [1],utilizing pre-processed CT images to eliminate clutter to improve average precision, and CNN for categorization at different levels [3], using histological pictures to develop an approach for colon cancer recognition and categorization using CNN [5], using the most up-to-date region-based CNN technique to detect polyps in colonoscopy screening images and videos [6] etc. On the other hand, Federated learning is a training ground that aims to solve the issue of information governance and security by collaborating on training rather than data transmission. Even though not all technological concerns have been settled, FL will undoubtedly be a topic of research in the upcoming years. But many researchers have already addressed some research on this topic. In particular, using mammography characteristics as a classification approach developing the use of FL for multi-institutional collaboration by holding patients data privacy [10], developing a breast lesions classification system based on mammography features in order to assess FL's effectiveness in the real world [11], demonstrating a federalized future for healthcare technology,by pointing out benefits and influence of FL for clinical implants, as well as highlighting key factors and challenges in implementing FL for healthcare analytics etc. We will describe some of the study, its technique, objectives, outcome, and constraint in this section of the publication.

Jha et al. conducted robust and large-scale experiments for detecting, finding, and sectionalizing polyps using a popular dataset known as Kvasir-SEG, and showed that different algorithms perform differently on polyps of varying sizes and picture resolutions [1]. They favored the ColonSegNet model for detecting and localizing polyps, as well as the IoU and FPS metrics for detection and segmentation techniques. Furthermore, the qualitative results distinguished the failed cases. According to the qualitative findings, YOLOv4 with Darknet53 was previously demonstrated to be the best model for recognizing and locating polyps, however Ponugoti et al. showed that RetinaNet with ResNet101 outperformed them [12]. Colon SegNet is significantly faster than UNet-ResNet34. It is 4 times faster in terms of processing colonoscopy frames and 11 times faster in terms of processing

speed than DeepLabv3 with ResNet101 [13]. Inception Resnet is unambiguously edified with complete image frames. This may cause a temporal delay, but it provides a conventional colonoscopy detection. Research in the future will be forced to focus more on the development of improved algorithms for polyp recognition, localization, and deconstruction tasks, and the quantity of parameters must be addressed when developing models. Furthermore, difficulties like saturation, reflectance, bubbles, contrast, and others should be addressed more efficiently while identifying, localizing, and segmenting polyps [14]. With the help of data augmentation, calibration, and other advanced methodologies, this model can be used not only in clinical settings for endoscopists, but also in scientific computation sectors with further progress in terms of accuracy, precision, and rate of acceleration.

Godkhindi et al. exploited pre-processed CT images to remove clutter, which enhanced average precision, as well as CNN for classification at various levels [3]. The colons were classified as Type 1, Type 2, and Type 3 to assist eliminate secondary noise. The second group of colon blocks is prone to low sensitivity and specificity, which is why the images were further processed before the polyp was identified. Following that, KNN (K Nearest Neighbor) and RF training and testing were undertaken (Random forest). While deconstructing the colon, the precision levels for CNN, RF, and KNN were 87%, 85%, and 83%, respectively. For CNN, RF, and KNN, the vulnerability level of polyp detection was 88%, 80%, and 83%, correspondingly. As a result, the research shows that CNN classification outperformed KNN and RF in both colon segmentation and polyp identification. The use of a GPU in CNN processing can help to speed up the process. The recommended method can be used to investigate intracranial tumors, lung cancer, breast cancer, and a variety of other clinical imaging diagnostics. While most cancers can only be cured if caught early, accurate detection of precancerous tissue expansion with computerized tools can assist patients in receiving the proper treatment.

Kwak, M. S. et al. established a methodology in which CNN was employed for colon cancer recognition and categorization using histological images [5]. By analyzing histology images and applying CNN to uncover patterns based on the development of Lymph-Node Metastasis, this technology may detect and classify the stages of cancer patients. A dataset was extracted from 600 test subjects and employed in this process after the machine had been trained. Imperfect photographs were filtered out of the test during pre-processing. Rectal cancer photographs were also eliminated because the outcome is different from that of colon cancer. To eliminate any irregularities or corruption in the test photos, Microenvironmental Feature Extraction was used to improve the procedure.

Shin et al. adopted the latest region-based convolutional CNN technology to find out polyps in pictures and videos obtained from colonoscopy screenings [6]. Automatically classifying colon cancer from polyp pictures is difficult due to the difficulty of recognizing abnormal tissue growth. Colon Polyp images are unfortunately in low supply. Because of brightness and color difficulties, many Polyps may not be recognized to train their model. They applied an Inception Resnet transfer learning approach to train their model .To increase the quantity of images of polyps, an augmentation was used to solve the problem of a Colon polyp shortage. Flipping, rotating, shearing, cropping, zooming in and out, and other augmentation techniques changed current data to produce new polyp images. Regional Proposal Network was used to classify polyp regions. It takes input as images

and shows the number of polyp regions as output. Furthermore, to detect colon polyps from video frames, an offlearning scheme was introduced to make their model more efficient. Moreover, they utilized false positive learning techniques to reduce false positive outcomes. After performing Haar, LBP and offline methods, the F1 scores were 40.8%, 38.5% and 86.9% resepectively. Although the detection system is not good for real time polyp detection as the model processing time of each frame is time consuming, but as time is not a constraint, this model can detect polyps efficiently.

Angermann et al. proposed leveraging picture databases to develop an automated approach for detecting polyps [7]. Detecting polyps in real time can be improved by using video sequence analysis. A new spatio-temporal coherence module and emulsion of feature descriptions have been included. Cascade Adaboost learning and active learning methods were used to create the classifier, which gave it more negative new cases. They used an image database containing six patches of picture: one positive patch with polyps and five negative patches with no polyps to train their classifier. The presence of polyps and a set of Regions of Interest are detected by the Classifier polyp detection function (RoI). The non-polyps regions of interest (RoIs) are then sent back into the classifier for an active learning process to achieve the best results. The fundamental issue with the still footage-based method of video analysis is that it does not take into account the information from previous frames while generating the new output. As a result, this procedure does not produce consistent results. To tackle this, they examined the Region of Interests of polyps in two prior frames before giving output in their classifier and matched if the previous frame's ROIs overlapped with the current frame's. If the current RoI matches the previous RoI, the output is kept; if the RoI does not match, the frames are discarded. This is the method of spatiotemporal coherence. They employed the LBP and Haar methods in their classifier, with the LBP scoring 20.24 percent F1 and the Haar technique scoring 31.10% F1.They got higher accuracy by using the Haar method in the classifier.

Bae et al. discussed how learning-based detection is difficult to detect polyps because the data set is imbalanced, and detection gives us a biased result because learning-based classification requires a large amount of data, and samples without polyps are greater than polyp samples [8]. As a result, they introduced a data sampling-based boosting framework in this paper in order to detect polyp from an unbalanced data collection. They employed up/down sampling in their paper and used this sampling approach to construct a polyp detector. Furthermore, they applied partial least square analysis as a learning method to strengthen the unique distinction between polyp and non-polyp that have similar appearances. Additionally, they employed certain complicated datasets to demonstrate the efficacy of their theory. They also employed a formula for precision (Precision = TP/(TP+FP)) and recall (Recall = TP/(TP+FN)) while analyzing the results. They indicated that learning-based detection can be utilized to detect, track, and classify complex objects using this sampling strategy.

Riberio et al. applied CNN to detect acid gastric polyp. This is a novel strategy that we discovered while reviewing all of the thesis [15]. For polyp identification, they used the SSDGPNet architecture and SSD for Gastric Polyps. The authors of this article repurposed data that had been abandoned by Max-Pooling layers in order to maximize the precision of the feature maps' data from the feature pyramid. They concatenated the data from the pooling layer as an extra feature so that it can aid in the categorization and

recognition of the polyp. In the meantime, to successfully expand the number of channels unfolded from the top layer in order to establish an explicit relationship between the layers. Using improved SSD for gastric polyp, real-time polyp detection is possible at 50 frames per second that improves mAP from 88.5% to 90.4%. The trial also revealed that SSD-GPNet had a high detection polyp recall of over 10%, especially when detecting tiny polyps. Previously, polyps were detected using elliptical characteristics, texture, color, and location, but it was time intensive and the method had a high percentage of false positives, according to this article. As a result, the authors proposed that the polyp can be detected in real time using SSD architecture. They used 215 patients' pictures with gastric polyps to conduct the experiment to evaluate the performance of SSD-GPNet. After collecting the data, they ran the algorithm and acquired a precision of 93.92% in SSD-GPNet. So they showed in the research that SSD-GPNet can detect more true positive polyps and is very promising in detecting gastric polyps, but there are certain limitations to this SSD-GPNet. Because it contains numerous parameters, the time output is slightly reduced, and the authors of this paper merely concatenated pooling results to expand feature maps to avoid picture data loss due to Max-Pooling.

Sirinukunwattana et al. recommended a procedure to detect and classify nuclei in H&E stained histopathology images of colorectal cancer following SC-CNN and NEP associated with Softmax CNN respectively [16]. In this procedure, SC-CNN reverts the probability of a picture element existing in the midpoint of the nucleus and detects all nucleus in an image by discovering their center point, despite their labels for classes. On the other hand, NEP along with CNN is used to classify the identified nuclei more precisely. For this paper, a large data set consisting of almost 20,000 explicated nuclei of four different classes from colorectal cancer images are used. One of the many factors which obstruct the techniques that are automated for detecting and classifying all cell nuclei is low-grade picture standard that could arise as a result of a defective fixing. Moreover, low quality staining while processing the tissues and autofocus omission throughout the computation of the slide might be the reason as well. Besides, in colorectal cancer the epithelial nuclei sometimes have asymmetrical chromatin consistency and remain strongly concentrated concurrently with an unclear lining,so It becomes tough to distinguish a single nucleus. Another thing which made the classification strenuous was different looking but similar nuclei both in the same and other samples. For more error-free classification they used NEP along with softmax CNN which hold all related patch-based divination where the nuclei was being classified rather than its single patch based correspondence. Also it does not require the tough part of nuclei segmentation that might be challenging for the reason stated beyond. This proposed method used small patches regardless of an entire image which increases training data that is important for all CNN. Again, it helped to confine small nuclei in images.

Liang et al. presented a method for recognizing colon cancer utilizing histopathological pictures of colon lesions using Multi-Scale Feature Fusion based CNN (MFF-CNN) based on shearlet transformation [17]. It is a process for extracting features automatically. The distinction between this technology and others is that the system fetched histopathology images as well as extracted secondary features. Shearlet coefficients were used in numerous iterations to extract secondary properties from the source image. This method has an F1 score of 0.9594 and a 96% accuracy rate. False positive and false negative rates can be lowered to 2.5% and 5.5%, respectively. Furthermore, this discovery has the potential

to provide real-time and accurate colon cancer detection.

Micah et al. established the use of federated learning for multi-institutional alliance, allowing machine learning models without any need to share patient information [10]. They evaluated federated learning to two possible cooperative technique approaches, IIL and CIIL, and discovered that neither can equal federated learning's efficacy. Large volumes of data are required for neural network models for linguistic classification of images. Obtaining adequate details in the diagnostic imaging sector is a serious difficulty. Coordination across organizations might help to solve this problem, however sending clinical information to the server site is fraught with regulatory and data control issues, particularly amongst multinational organizations. By repeatedly pooling locally competent networks to a central controller, they used FL towards the BraTS dataset to create an efficient classification algorithm that detects the diversity over several institutes without revealing any patient information. The analytical findings showed that federated semantic classification methods, Dice = 0.852 performed similarly to models trained via sharing information, Dice = 0.862 on intermodal neuroimaging. They used a deep learning model, CNN framework, U Net, to perform supervised feature extraction. By contrasting FL to two other collaborative deep learning they demonstrated that IIL performs relatively poorly to FL and CIIL, and that CIIL is less stable and difficult to evaluate than FL. The main drawbacks of IIL are decrease in efficiency as the institute grows and the problem of catastrophic memory, in which accumulated characteristics are lost when retraining information replaces the old data. With a large range of scores after each cycle, CIIL is less consistent because of CIIL's unpredictability. However, Data for training assessment throughout CIIL and FL learning, DC results for organization 0 suggest that the CIIL models suffer from some catastrophic memory. Forgetting could be the source of CIIL's inconsistency. Moreover, In any of the above learning settings, organizations could be included or withdrawn after some training. For such instances, the empirical framework through additional collaborative learning should be a common lineage to the model that is determined through training from the start with the new batch of collaborators. Any discrepancies would ultimately be discarded and a new dataset could be established, according to the limits identified in this study of the specific collaborative arrangement. However, even with imbalanced datasets, such as the real BraTS, their FL trials reached 99 percent of the prediction accuracy of a data sharing approach.

Holger et al. did a study to construct a breast lesions classification method using mammography features to analyze FL's efficiency in the physical world [11]. In a genuine shared scenario, they studied the application of FL to create diagnostic imaging classification techniques. They demonstrated that they can effectively train artificial intelligence systems in federation despite major variances in datasets among all locations and without standardizing information. The findings showed that models generated with FL outperform trained models only on an institute's local data by 6.3% on average. Additionally, when compared to the test results from the other participating sites, the models' generalization improves by 45.8%. Moreover, they used the Federated Averaging method in their tests. A softmax cross-entropy loss was used in this study, which is typically used for inter classifiers. They concentrated on demonstrating how FL functions in a collaborative training environment. The FL architecture is written in Tensorflow8 and also to unify learning setup. On a clinical level, every client divided their information into train, test and validation sets. Results from all photos from a patient's condition were averaged

with each other at close intervals to get a patient model's predict. In the FL context, they attained a performance level of 0.68 for model parameters, indicating FL's capacity to produce designs equivalent to those learned when data is collected in a centralized system. They employed 1e-4 starting training data, step-based learning rate depreciation, adam optimization and model loss tangent for every client. However, they purposefully avoided using data harmonization procedures to examine the impact of diverse data categories in this investigation. Besides they left future work for histogram equalization and other techniques to harmonize non-IID data across several sites, or look into built-in domain signaling pathways inside the FL architecture. Likewise, in their FL approach, they did not adequately address issues related to data size variability and category balancing. They also didn't try any privacy-preserving strategies to limit the likelihood of model flipping and data leaks depending on the different classifiers. In spite of these obstacles, they were able to train mammogram classifiers in a real-world FL context that outperformed locally trained models without the necessity for centralized datasets.

Nicola et al. visualized a federalised prospect for healthcare technology, and with this perspective paper, they shared their prevailing opinion in order to provide clarity and description for the public about the advantages and influence of FL for clinical implants, along with highlighting important factors and difficulties in implementing FL for healthcare analytics [9]. The distinctive features of FL in medicine also create difficulties, such as ensuring information when interacting via redundant nodes, developing safe encryption technology to preclude security breaches, or implementing suitable node scheduling algorithms to create the correct use of dispersed mobile computing and reduce idle time. Although this is true that all machine learning-based technologies, mechanisms developed in a federated method may be able to produce even less skewed conclusions and greater responsiveness to unusual occurrences because they were probably subjected to a more full data dispersion. Implementing FL on a worldwide scale would ensure that healthcare practitioners are of good quality irrespective of surgical site.FL may lessen the barrier towards becoming an information provider by assuring clients that their data will remain at their own facility and that data access can be canceled. Moreover, according to their research hospitals and clinics can maintain total ownership and administration of their client records, with comprehensive data usage transparency, reducing the danger of third-party exploitation.The collaborating organizations in a FL procedure with an aggregating gateway may be entirely unknown to one another. Nevertheless, it has been demonstrated that the systems may retain knowledge given specific circumstances. As stated by the author, mitigation strategies such as restricting the resolution of information and introducing interference as well as guaranteeing proper differential privacy, may be required and are still being researched. Additional issue is that data heterogeneity may result in a scenario where the global ideal option is not the best for a single local institute. To further strengthen security in a FL context, strategies such as differential privacy. Overall, the community has shown enthusiasm in FL for medical applications and FL methods are a burgeoning field of studies. Despite these benefits, FL does not address all of the challenges that come with learning from medical information. Quality of the data, biases, and standardization are still important considerations in model development.

Chapter 3

Methodology

The purpose of our CNN based colon cancer classification model in deep learning is to detect cancer through analyzing and classifying histological images of human colorectal cancer. The pre-eminent step of our research is to procure a proper and appropriate dataset containing both cancerous and non cancerous colon tissue images. The following step is quite significant to acquire an uncorrupted and non biased result where we had to process the images in the most suitable way. Then our dataset has to be divided into train, test and validation data. Before feeding the dataset of images into a convolutional neural network, we have to make sure that all the images are of the same size. Classifying images is a major step and the machine learning classification task that consists of more than two classes, or outputs is known as Multiclass classification. The softmax function can generate a classifier through being tacked into the last layer of a neural network. It creates an output vector of length K through a probability distribution over K classes. The probability that the input corresponds to the corresponding class is represented by each element of the vector. The most likely class is determined by selecting the highest probability vector's index. Convolutional Neural Networks (CNNs) perform adroitly on multiclass classification tasks, especially for images and text, aside from the softmax function. A CNN collects important features from data, especially those that are not affected by scale, transformation, or rotation. This allows it to detect images that have been rotated, resized or shifted off-center, permitting it to perform image multiclass classification jobs with greater accuracy.

3.1 Dataset Description

For constructing precise and vigorous Deep Learning models, large medical dataset with raw images is a major prerequisite. Even though there are many medical image datasets available, our target dataset must carry various kinds of medical entities, especially cancer pathology. Appropriate image datasets for Machine Learning are quite infrequent and privacy concerns restrict access to these data.

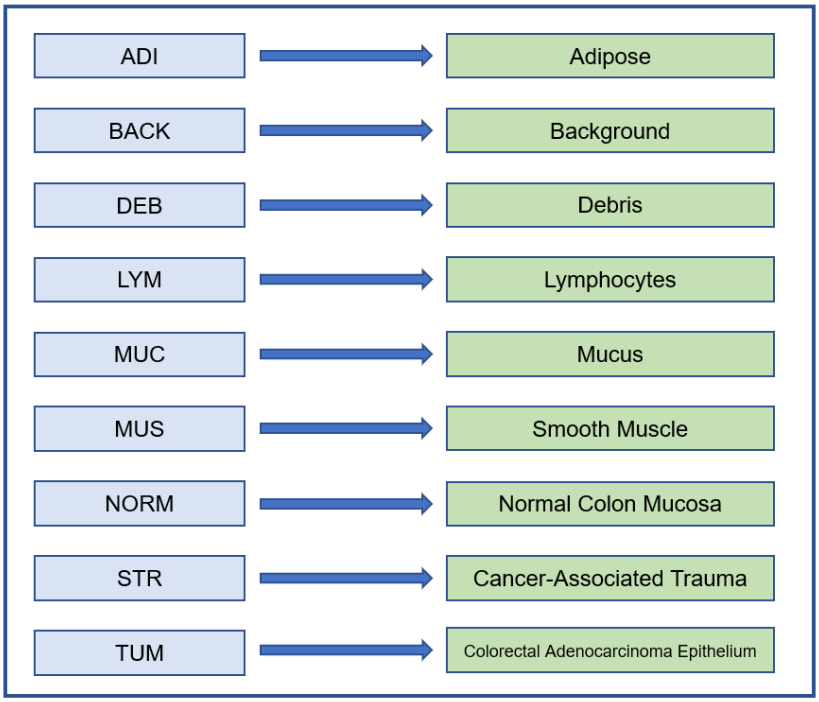


Figure 3.1: Class Description

After exploring plentiful datasets, we have chosen to work with the one of Kather et al. that contains a set of 0.1 Million non-overlapping histological images of human colon cancer and healthy tissue [18]. The dataset comprises 9 classes described in figure 3.1. The images of tissues were gathered from 86 H&E stained tissue slides of human cancer from FFPE samples from the NCT Biobank and UMM pathology archive. Normal tissue classes were augmented with tumor free sections from gastrectomy specimens to improve diversity. Samples of the tissue included CRC primary tumor slides and tumor tissue from CRC liver metastases.

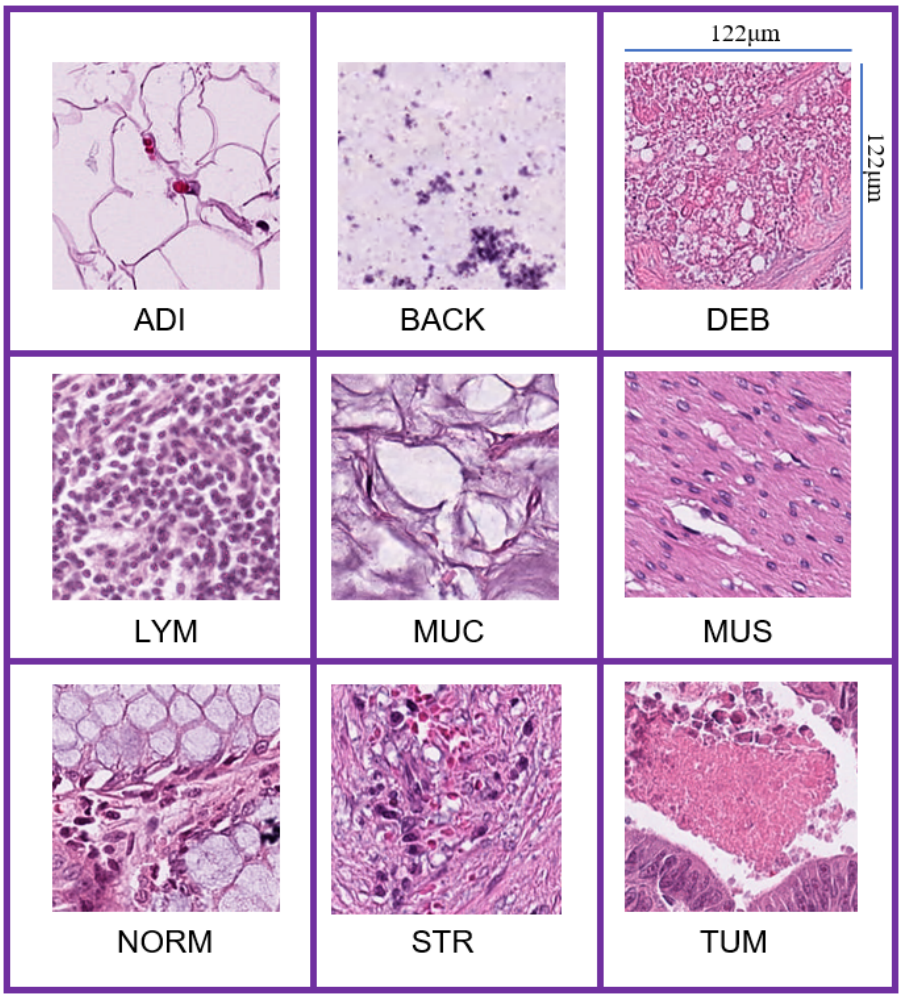


Figure 3.2: Tissue Image Samples

The images attached to our dataset are 224x224 pixels at a rate of 0.5 microns per pixel. Using Macenko's method, all the images are colour-normalised [4]. The samples of these tissue contain both salient tumor slides and tissue from CRC liver metastases as exhibited in figure 3.2. For increasing variability, non-tumorous regions from gastrectomy specimens are supplemented with normal tissue classes.

3.2 Image Pre-processing and Distribution

As high-resolution photographs have a high number of pixels, they necessitate more processing power. Furthermore, the image resolution may differ throughout the training phase and lead to an inaccurate result. Filtering high-resolution images in a convolution layer takes longer than filtering a lower-resolution image. The quotidian way of normalising images is to reshape them which is the first step of the whole processing expedition. In the case of image datasets, dividing all the pixel values by 255 is a cardinal step as the minimum pixel value is 0 and maximum 255. To bring a reduction in the amount of parameters at the Conv layer is essential. Shifting the height and width, enabling the horizontal flip will convoy variations in the training data while leading towards a robust and explicit result. The dataset is splitted into training, testing and validation with the ratio of 7:2:1 to prevent overfitting as displayed in figure 3.3. Random sampling is used for the split with proper caution to avoid the creation of any imbalance between the training and testing split. All the models are fetched through keras API and implemented using the tensorflow library.

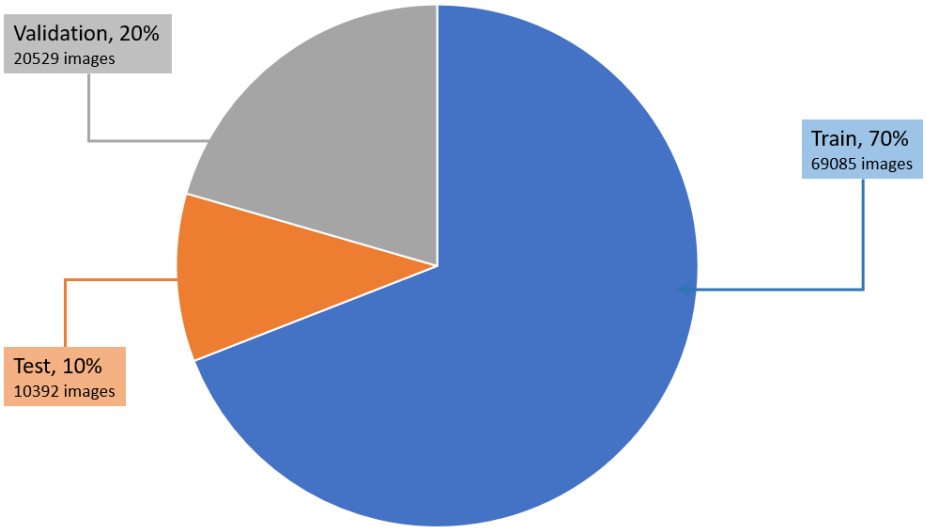


Figure 3.3: Image Directories and Distribution in CNN Models

For the implementation of FL, we propose a slightly different distribution structure than the one of CNN model. As we are focused on privacy, we do not test our models after each epoch using the validation dataset and the whole dataset is divided into two different directories, 80% to train directory and 20% to test directory. This is illustrated in figure 3.4 Images in train directory would be used for the general purpose of training our local clients. When the whole model is trained, we would use test directory to project a prediction model based on the confusion matrix. Moreover, to retain simplicity we would not use any data augmentation for our FL dataset. But resizing into (50×50) pixels would be the optimal option to feed our dataset into our FL model.

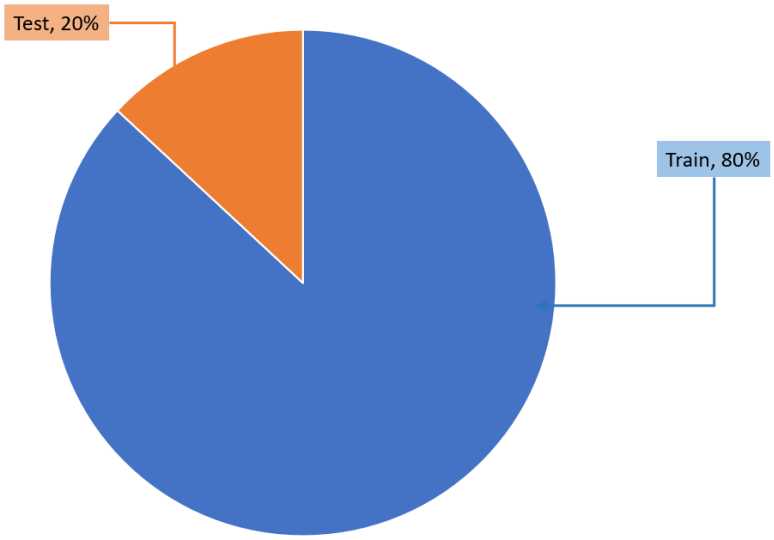


Figure 3.4: Image Directories and Distribution in FL Model

3.3 Convolutional Neural Network Architecture

Convolutional neural network is eminently illustrious Deep Learning model motivated by biological processes that inspired the connectivity pattern between neurons. Being a multilayer perceptron and shared weight network composition, it is extensively utilised for a flawless detection of cancer with a strong recognition propensity and formidable prediction accuracy [19].

CNN is constructed of neurons with learnable weights and biases. Having appropriate qualities for image processing make it easier to build the forward function and deduct the amount of parameters in the Neural Network. The neurons or perceptions in most neural networks accept some inputs, apply a dot product operation with weight, and arbitrary follow it with non-linearity. As an output, it produces a scoring function and a loss function.

ConvNet is a series of layers arranged with three dimensions: width, height, and depth, each of which uses a differentiable function to translate one volume of activations to another. From the initial pixel values to the final output, ConvNets alter the actual image through each layer. The Conv layer, being a series of learnable filters, is not only the central component of CNN but also it is responsible for the majority of the computational effort. In terms of picture input, the first layer filters may be used to detect all edges, while the second layer filter could be used to recognize various geometric shapes. CNN calculates the dot product between the filter's values and a specific place in the input volume by sliding each filter across the width and height of the input volume. Since the filter slides throughout the entire height and width of input volume, the Conv layer produces an activation map of 2D array that covers the particular filter responses at each location as demonstrated in figure 3.5

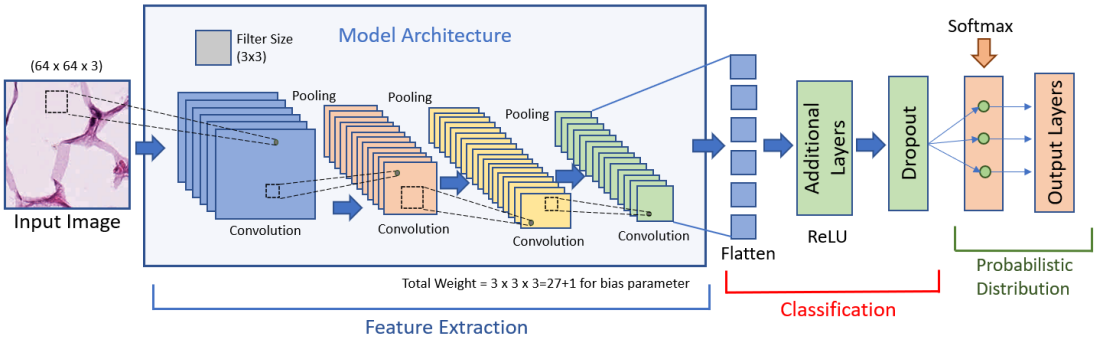


Figure 3.5: ConvNet

For output operation, depth, stride and padding are the three main factors. Depth and stride decides the number of parameters and filter movement towards a specific side respectively. Hence a smaller output volume is achieved. As shown in figure 3.6

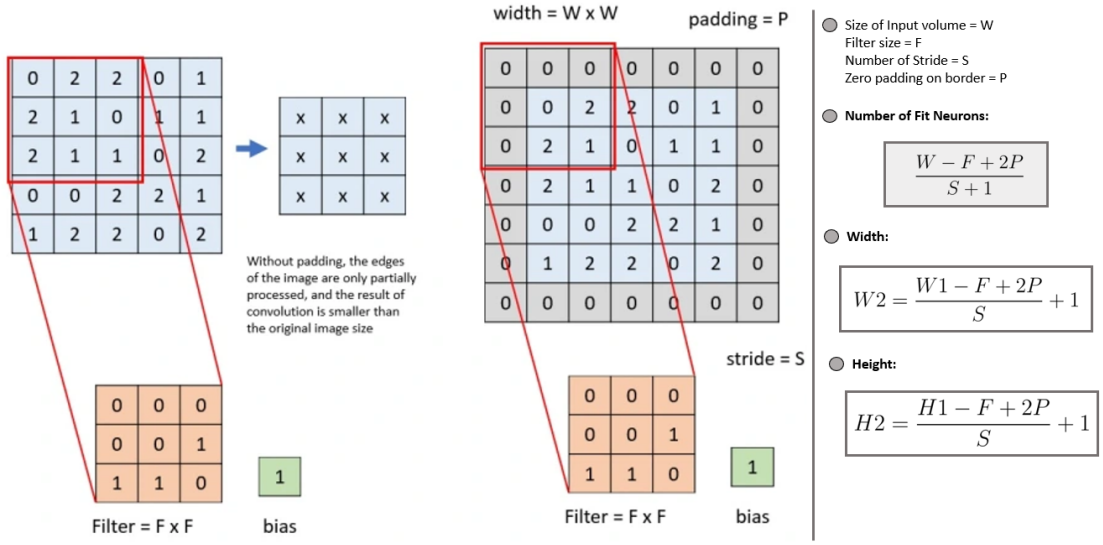
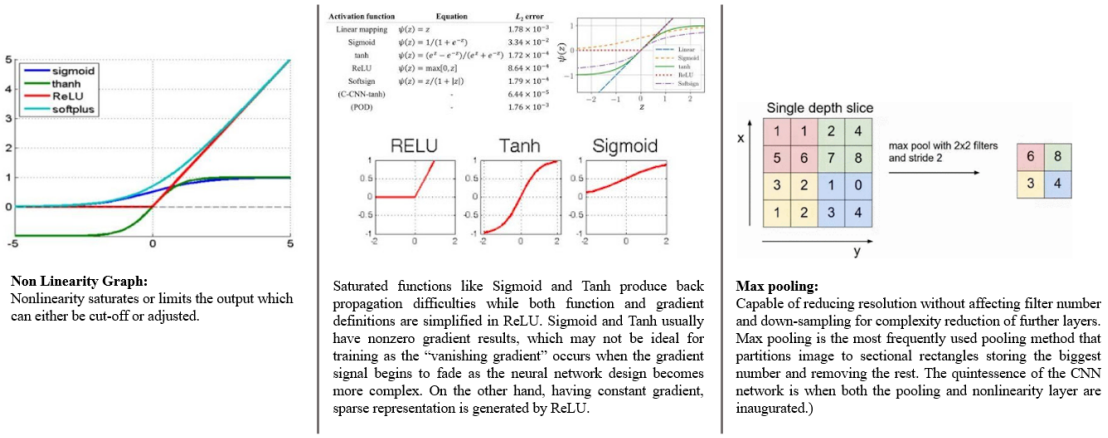


Figure 3.6: Output Operation of CNN



Non Linearity Graph:
Nonlinearity saturates or limits the output which can either be cut-off or adjusted.

Saturated functions like Sigmoid and Tanh produce back propagation difficulties while both function and gradient definitions are simplified in ReLU. Sigmoid and Tanh usually have nonzero gradient results, which may not be ideal for training as the "vanishing gradient" occurs when the gradient signal begins to fade as the neural network design becomes more complex. On the other hand, having constant gradient, sparse representation is generated by ReLU.

Max pooling:
Capable of reducing resolution without affecting filter number and down-sampling for complexity reduction of further layers. Max pooling is the most frequently used pooling method that partitions image to sectional rectangles storing the biggest number and removing the rest. The quintessence of the CNN network is when both the pooling and nonlinearity layer are inaugurated.)

Figure 3.7: Activation function and Pooling layer

3.3.1 VGG16 & VGG19

The utilization of VGG16 and VGG19 architecture lies with detecting and recognizing cancer cells from the images included in our dataset. Here, the 16 represents the layered weight and. All layers in VGG use the ReLU activation function. One of the advantages of the VGG16 algorithm is that it uses smaller receptive fields (3×3, 1 stride) compared to other algorithms. Remarkable criteria of the VGG-16 is that instead of retaining so many hyper parameters we will use a much simpler network.

VGG19 model provides a 19 layered weighted structure which is its only exception from VGG-16. Among those three layers, the first two carries 80% of the total channels while the third has only 20% channels, one for every class. ReLU is used by all the concealed

layers in the VGG network. The smaller layers benefit from having a larger number of weighted layers, thus increasing the performance of and accuracy of the system.

3.3.2 Inception V3

Compared to VGG, Inception Network is computationally adaptive to a greater extent in terms of the amount of parameters originated by the network and the monetary value of resources including memory. To enhance the Inception network without losing computational advantages, the progressive, speedier and deeper network InceptionV3 includes various methods such as smaller or asymmetric convolutions, dimension and grid size depletion, and parallelized computations as shown in figure 3.8. It is built up with 42 layers while having a higher accuracy. To propagate label particulars beneath the network, it utilises Label flattening, Factorized 7×7 convolutions, and Auxiliary Classifier along with the utilisation of batch normalisation [20]

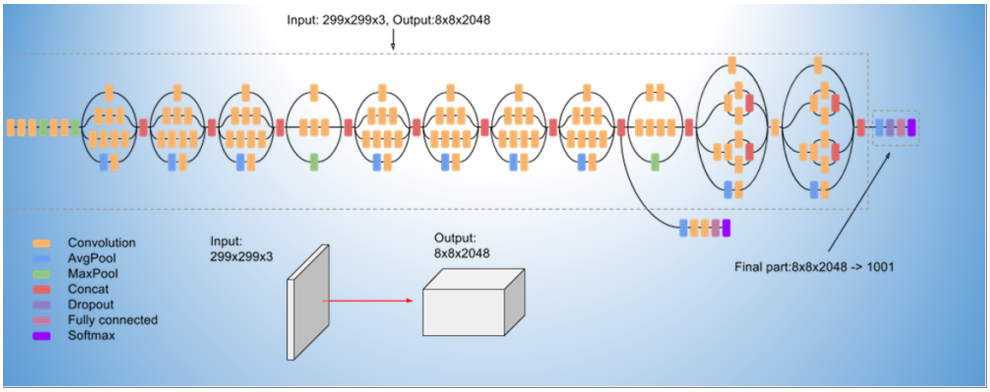


Figure 3.8: Inception V3

3.3.3 ResNet50

Residual Network is a deep network that is capable of obviating both the gradient diminishing and degradation issue by applying the residual mapping technique. The most frequent variety is 3 layered deep ResNet50 model that comprises 49 convolutional layers and a sole FC layer sustaining 25.5 million network weights. Its computational complexity is lower than that of VGG, even with enlarged depth. Feedforward neural networks with detour connectivity can be used to realize the calculation of the output xl displayed in figure 3.9.

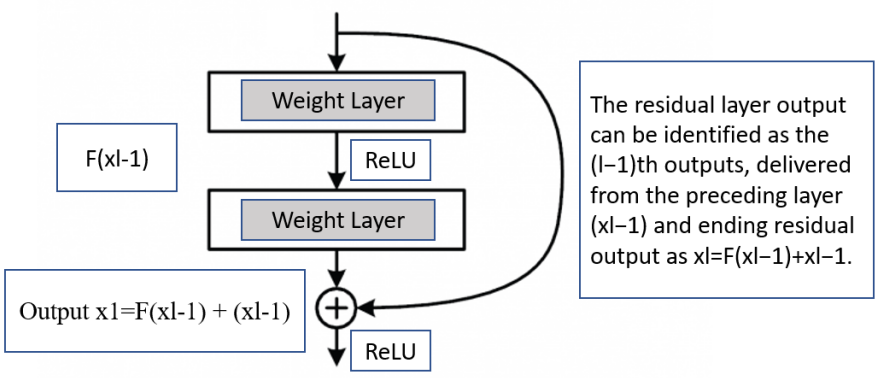


Figure 3.9: Residual network building block

ResNet ensures the simplicity of regulation and acquires increased accuracy as the network depth rises, resulting in more precise outcomes than the earlier networks.

3.3.4 ResNeXt50

ResNext is a new generation of deep residual network and an intensified variety of the Inception Network. It unmasks a new dimension in contrast to Resnet50 while following the same construction with some modifications. It possesses an architecture that has a multitudinous branch and is homogeneous with some hyper-parameters settings called cardinality that enables the rearrangement, split, and integration topology in an untroubled and constructive way [21]. Moreover, ResNeXt not only controls the resources more effectively but also escalates the retaining potentiality of the conventional CNN. ResNeXt50 is barely fallible when the cardinality is towering and performs admirably in comparison to ResNet.

A rudimentary comparison between the ResNeXt architectures is demonstrated in figure 3.10 from where we can claim that, even though the number of parameter is higher in ResNet50(25.5×10^6) than ResNeXt50(25.0×10^6), the FLOPs value(floating point operations/second) is higher in ResNeXt50(4.2×10^9).

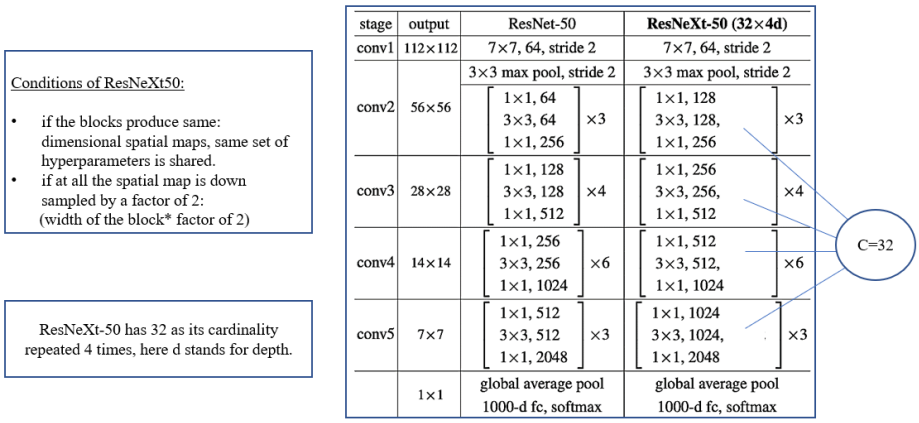


Figure 3.10: Comparison between ResNet50 and ResNeXt50 architecture

In a nutshell, the input is split by a network unit, modified into a requisite format and merged to acquire the result where the same topology is pursued by all the blocks.

3.4 Federated Learning

Collaborative training towards a decentralized model with privacy-preserving has attracted the researchers over many years. The advent of federated learning technology enables the training of a model with the incorporation of a central server while keeping the training data decentralized in the distributed clients. It is our goal to use FL to uphold the concealment of user data and bring an expansion in data. Concurrently, it permits the participants to collaboratively train a global model without sharing one another's private data as presented in figure 3.11. The regional data requires to be pre-processed for each contributor, which incorporates modification, digitalization, and standardisation for transmuting the original data into a standard arrangement along with distinctive privacy. Images of our dataset are distributed among different clients by dividing the total Size of Image with the number of clients [20].Uniformly dividing the dataset, our IID data is generated.

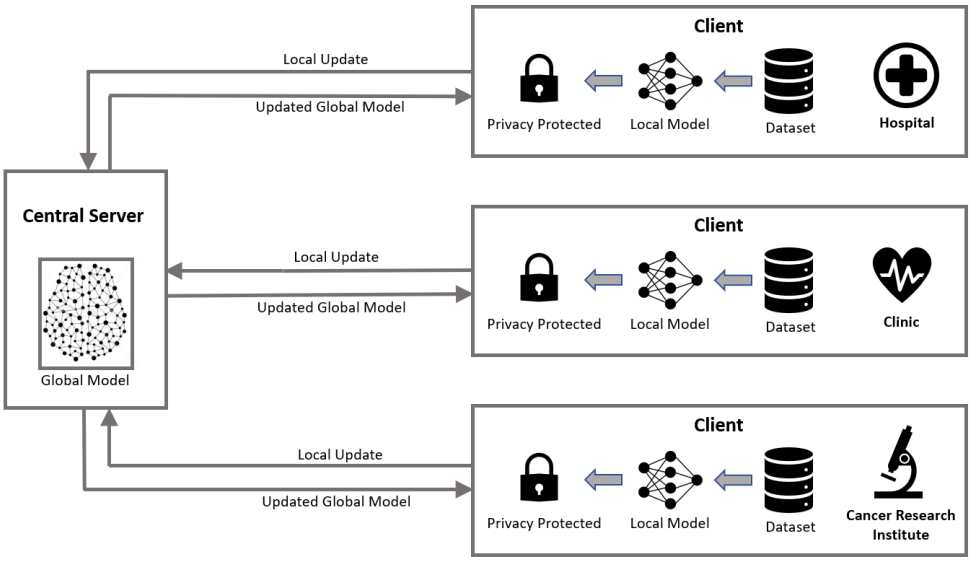


Figure 3.11: Visualization of Our Proposed FL Model

Once the model parameters transmitted by the clients are acquired, the server will recapitulate it according to the formation of the central server, upgrade the parameters of the current model, and persevere it for the next round of training parameter upload and collection to the participants before remetting it [22]. The rudimental FL iterative procedure is followed by the iterative process of our overall model, where we combine a CNN adapted to cancer tissue data samples and customize the model to form continuous iterations. Our proposed framework for federated learning is exhibited in figure 3.12.

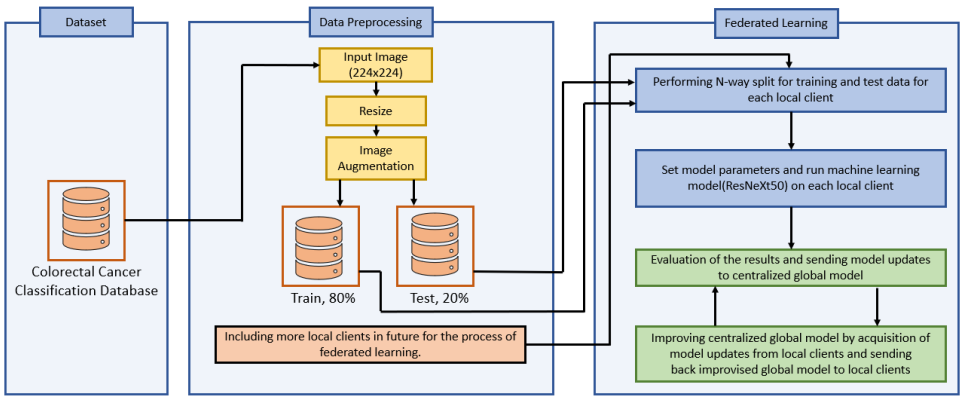


Figure 3.12: Proposed Framework for FL

In dispersed Machine Learning, the standard method of accumulating models implies that all participants have the same amount of training samples. In Federated Learning, members often possess uneven amounts of data. The local models are aggregated to solve this by weighting each local model by the amount of available training samples as depicted in figure 3.13. As a result, models with a larger number of samples are taken into consideration than those with only a few. This method appears to be uncomplicated but has shown predominance and efficacy in Federated Learning scenarios.

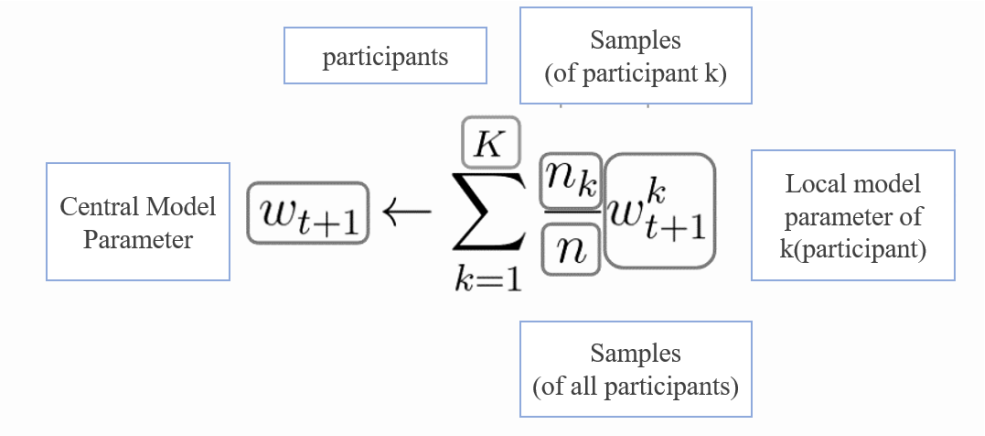


Figure 3.13: Federated Averaging

Each iteration of this process is known as a federated round, which includes concurrent training, update aggregation, and parameter distribution. The main parameters to control the calculating endeavour of FL are:
C = Clients or contributors taking part in an update cycle (in %)
E = Number of local epochs perpetrated by each contributor
B = Minimum batch size used for each local update
$\beta 1$ and $\beta 2$ = Hyperparameters

A certain amount of local epochs is run by the clients in the step of regional model optimization, Adam optimizer exploiting [23] the first and second order moments for beating the local minima:

$$\mathbf{w_{i,t}} \leftarrow \mathbf{w_{i,t}} - \eta \frac{\sqrt{\mathbf{1} - \beta_\mathbf{2}^\mathbf{n}}}{\mathbf{1} - \beta_\mathbf{1}^\mathbf{n}} \times \frac{\mathbf{m_{i,n}}}{\sqrt{\mathbf{v_{i,n}}} + \sigma} \tag{3.1}$$

Decaying averages($m_{i,n}$ and $v_{i,n}$ where, n = timestep index of the Adam optimizer):

$$\mathbf{m_{i,n}} \leftarrow \beta_\mathbf{1} \mathbf{m_{i,n-1}} + (\mathbf{1} - \beta_\mathbf{1}) \nabla \mathcal{L}_\mathbf{1}(\mathbf{w_{i,t}}; \mathbf{b_i}) \tag{3.2}$$

$$\mathbf{v_{i,n}} \leftarrow \beta_\mathbf{2} \mathbf{v_{i,n-1}} + (\mathbf{1} - \beta_\mathbf{2}) \nabla^\mathbf{2} \mathcal{L}_\mathbf{2}(\mathbf{w_{i,t}}; \mathbf{b_i}) \tag{3.3}$$

For image segmentation problems like tumor segmentation, the number of local epochs E is usually kept below 2 while the batch size B is increased as much as possible to exploit all the parallel computations given by the GPU.

The aggregation stage is a significant part influencing the performances of the algorithm to a great extent is performed across a weighted average.
$S_i = |D_i|$= number of samples of each client
ϵ = Memory regulator of previous models to ensure a smooth,infrequent changes in weight

$$\mathbf{w_{G,t+1}} = \frac{\epsilon}{\sum_{j=1}^{N} S_j} \sum_{i=1}^{N} S_i w_{i,t} + (1 - \epsilon) w_{G,t} \tag{3.4}$$

Explainable Artificial Intelligence

The Explainable AI (XAI) appertain to elucidate the decision making process of machine learning models while maintaining an elevated prediction proficiency. It produces comprehending methods creating an opportunity for the users to count on and contrive the efflorescing generation of artificially intelligent partners [24]. Even though most machine learning models are utterly resilient like a black box, XAI is programmed to describe its purpose, rationale and decision-making process explicitly for a convalescent user experience like the visualisation in figure 3.14.

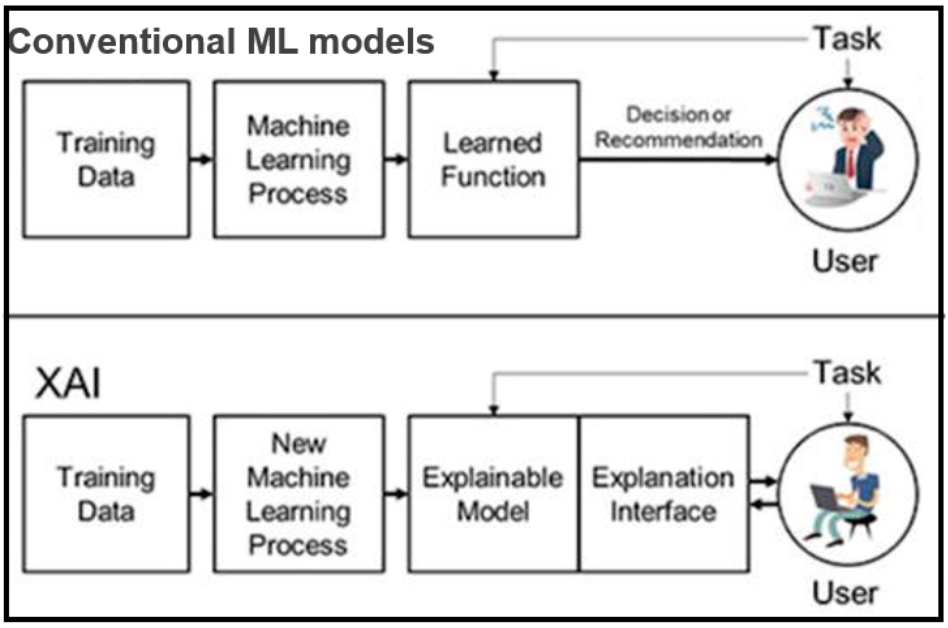


Figure 3.14: An average person can understand the purpose, rationale and decision-making process through XAI

Among many XAI techniques, Lime or Local Interpretable Model-agnostic Explanations can simply provide interpretable and local explanation of any black-box model. A prototype of the interpretability method of XAI is shown in figure 3.15. Generally, lime follows four steps including input data permutation by generating samples through super-pixels of image, predicting the class and calculating the weight of each artificial data point, fitting linear classifier to explain the most significant features. Implementing a pre-trained CNN model on XAI enables us to discern the prediction accuracy and analyse the model behaviour while classifying the image.

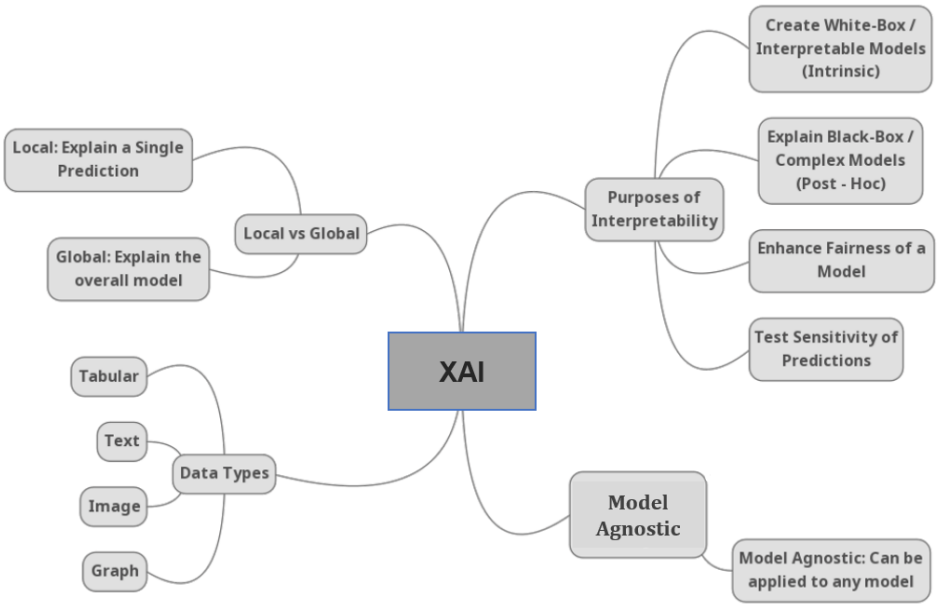


Figure 3.15: Interpretability of XAI

3.5 Workflow

The following diagram 3.16 illustrates the work flow of our research. After analyzing the feasibility of each our theorized models we are planning to implement using our aforementioned dataset. The division of images into different sub-categories and preprocessing steps are conducted following the diagrams. Then five different CNN models were trained and based on their trained model and prediction model results we will compile a comparison report shown in Results section. Based on the report, best CNN model which is ResNeXt50, will be used as primary model of FL implementation as shown in the right portion of the diagram.

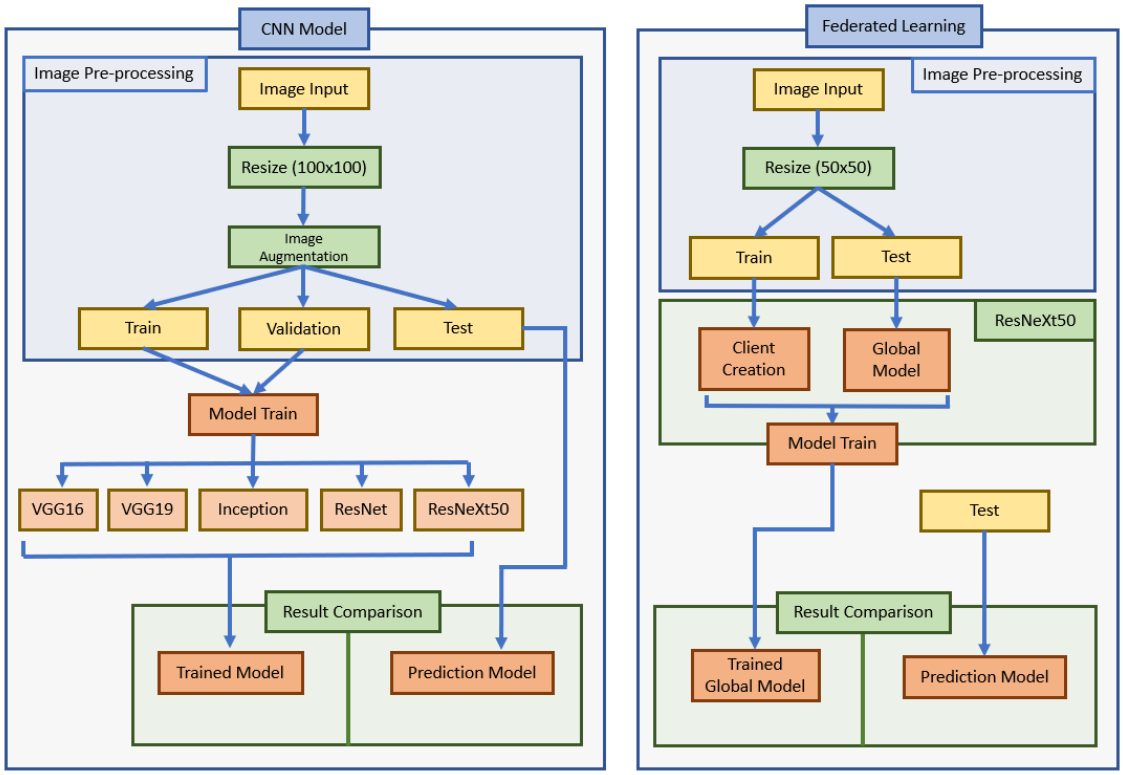


Figure 3.16: Workflow Diagram

Chapter 4

Implementation

We deployed our theoretical knowledge to the test by implementing a federated learning architecture utilizing existing CNN models after thoroughly comprehending the whole architecture of CNN and federated learning as stated in the methodology chapter. To compare and determine the best model for our federated learning application, we implemented five different CNN models which are VGG16, VGG19, InceptionV3, ResNet50, and ResNeXt50. The same configuration was used for all of the models. After choosing the best model for our federated learning application, we implemented it using that specific model. Extensive discussion of these implementations are discussed in this chapter.

Tensorflow and Keras:

For our implementation, we have used Google's tensorflow platform. Tensorflow is an open source platform for machine learning and deep learning applications. Besides, we have used keras which is a neural network API for computing deep learning and computer vision models. The models were pre-loaded from the tensorflow library. Layers like flatten, dense, and dropout were manually inserted after the model's own design before the output layer, which comprises 9 distinct neurons. As weight, ImageNet was used which is a large database of hierarchical classification images of over 1000 different classes.

4.1 CNN Models Implementation

This section explains the CNN models and it's implementations. It also includes the features and methods that were used to improve the overall result, reduce error rate and chances of overfitting in our model.

4.1.1 Image Pre-processing and Augmentation

Although our dataset consists of images of (224×224) pixels, we have resized the images to (100×100) to balance computational complexity for our application. After successfully integrating the train, test and validation images from our dataset into our model. We used image augmentation techniques to alter our train images. ImageDataGeneretor was used to apply these augmentation which are image resizing, shifting, vertical and horizontal flipping. The reasoning behind this was to introduce variation for our images. As a bonus, this method also helped us to reduce overfitting for our models.

4.1.2 Layer Architecture

After the general architecture and construction of layers in each model, we have added several layers such as flatten, dense and dropout layers. These layers were added before the final output layer which has 9 output neurons. This is shown in the following figure 4.1. As each of our models have different layer architecture, we have added additional layers to all of them. As our convoluted classification model is processed, the flatten layer transforms the data into a 1-dimensional linear vector as our output layers do not recognize 2 or 3 dimensional shapes directly. So the addition of a flatten layer creates a long linear vector of feature array. Dense layers with 1024 neurons were added after a flatten layer. Then a dropout layer was added which extracted and dropped unimportant features from our model. It further reduces the chance of overfitting. As our final layer, another dense layer was added with 9 output neurons. These represent our 9 different classes. This is the architecture that was followed in all our CNN model implementations.

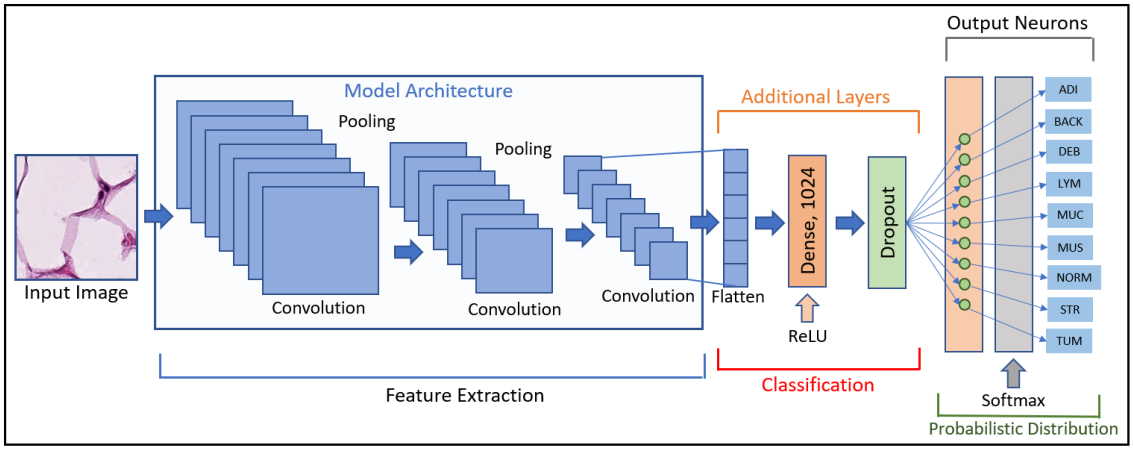


Figure 4.1: Layer Architecture for CNN Models

4.1.3 Earlystop: A Technique To Reduce Overfitting

Earlystop is a concept of keras library. The main use case of this method is to reduce overfitting in models. It has the capability to reduce overfitting the model despite not losing accuracy or effectiveness of that model. It is a cautionary technique that has been used in image classification in recent years. In our model, we used earlystop with a patience of 10, meaning that if our validation accuracy does not change in 10 successive epochs, the model will stop training automatically, thus reducing the probability of overfitting.

4.1.4 Prediction Model Generation and Confusion Matrix

Our CNN models generate predictions based on 10392 images which were used from test directory images that were not used during the training of our model. The images were taken as list items which represent 9 probable results based on our 9 different output neurons. The 9 probable results represent the probability of that image belonging to that particular class. After calculating the highest probability between 9 classes for each list, we form a prediction list of all test images.

The test images were labeled 0 to 8 to illustrate the confusion matrix. Then a test label list was created by appending the corresponding labels of each test image. By comparing prediction list and test label list, confusion is generated to calculate the result. In the confusion matrix, the diagonal values represent the true positive values of each class, whereas the other values represent faulty predictions.

Explainable AI Integration

A pretrained ResNeXt50 model was loaded to explain our images using the LIME library. We selected one image from each class of our training directory. All images were reshaped to 100×100 resolution and kept 1000 artificial data points which will be the same as the input image. Then we passed the selected images from each training directory to the LIME explainer function and it marks out different areas of the images. Results of these implementations can be found in figure 5.5.

4.2 Federated Learning Implementation

Implementations of the federated learning had been explained in the following section which would include the process of the data being trained in the local client devices and the aggregation of the local model in the central server.

4.2.1 Image processing and Resizing

We resized the images of our train and test directory from our existing dataset which consist of (224×224) pixels to (50×50) pixels to decrease the complications in our computation. The reason behind the application of the train and test directory is because images of the train directory would be distributed among the end devices/clients for local training of the data and the images of the test directory would be used for testing purposes in the global model.

4.2.2 Local devices (Client) Creation

In the architecture of federated learning, the datasets were trained in the local devices, Afterwards, the end devices would send their trained local models from the dataset to the central server for aggregation. Therefore, we created 10 local devices for our model which were denoted as clients, and the clients would train their model based on their own local datasets.

4.2.3 Distributing Data Among Clients

We distributed data among the end devices through data sharding which is a process of distribution of smaller datasets known as logical shards/chunks from a larger dataset. Although, in a real-world application, each client would contain different sizes and variations of the dataset, in our FL prototype we distributed the shards equally among the clients in the model, and the size of the shards were determined by the following formula:

$$Shard\ Size = \frac{Total\ Num.\ of\ Images}{Num.\ of\ Clients} \tag{4.1}$$

Following that, each of the data shards was processed and assigned to the clients through the process of batching. After the completion of the process, every client acquired their local datasets and they were ready to train their datasets.

4.2.4 Integration of ResNeXt50 and Its Configuration

As mentioned earlier the local datasets were trained in the local devices. We implemented the ResNext50 architectural model on each client to train their local data and in the global model for its testing purposes. We also implemented 3 extra CNN layers which were the Dense layer, Flatten layer, and Dropout layer. Similar to the CNN implementation we used ImageNet as it's the default weight in the model. Adam optimizer (lr=0.00001)

was used for optimizing the accuracy of ResNeXt50 and also we implemented categorical cross entropy for calculating its loss function.

Communication Rounds

We ran the communication round 20 times to attain a certain accuracy for our global model. Communication cycle/round where the clients containing the datasets that we created earlier would acquire their weight according to the global model's weight where each of the 10 clients would start to train each of their local data and produce their own accuracy of their local model. Afterwards, each of the clients sent their trained model to the central server, also known as the global model, for aggregation which is an averaging operation of the FL model. Following the first aggregation, the global model produced a new weight and in the next round, the clients would set their new weight according to the newly found global weight. This type of operation was implemented as a weight scaling factor in our model.

After the completion of the communication cycle, the images of the test directory were provided in the global model to construct a prediction model. Finally, we generated a confusion matrix using the prediction model.

Chapter 5

Results

Initially, five distinct models were trained over the identical dataset, as indicated in the implementation section. Precautionary measures were enacted to minimize the probability of overfitting or underfitting. Early-stopping was a significant preventative step. Consequently, our models did not train for static epochs or runtime. Rather, different models were trained for different epochs. Despite that, Initial results show no visible discrepancy or infectivity. Considering this, the result of all the models can be evaluated side by side. After compilation of each model, the following results can be derived from it- Accuracy Curve, Loss Curve, Scatter and Line model, loss and confusion matrix. Moreover, using a confusion matrix, we can determine the accuracy, loss, precision and recall value which can be used to calculate the F1 score of our models. The formula for calculating F1 score are given below-

Given:

$$TP = True\ Positive$$
$$TN = True\ Negative$$
$$FP = False\ Positive$$
$$FN = False\ Negative$$

Using this we can derive:

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{5.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{5.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{5.3}$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5.4}$$

We will explain unique results of each model in this chapter, then create a comparison graph and chart to distinguish between these models and their effectiveness. Furthermore, the results of federated learning and XAI will also be explained here.

5.1 Model Specific Results

We have accumulated thorough information from the results of each model. As ResNeXt50 has scored the highest accuracy among all other models, we would like to utilize it. Total epoch for this model was 39 until early-stopping was called. Over 39 epochs, this model managed to reach a peak accuracy of 0.9955 meaning 99.55%. The following figure 5.1 shows the comparison between training and validation accuracy where epochs are displayed in x-axis and y-axis provides the accuracy value. In addition, it shows the loss value for training and validation where in x-axis displays the epoch and y-axis corresponds with the loss value.

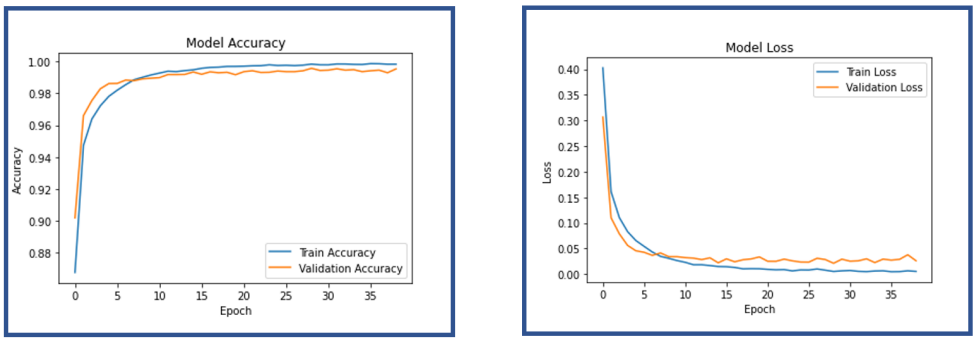


Figure 5.1: Accuracy and Loss Plotting

Figure 5.2 shows the scatter model for training and validation accuracy and loss over each epoch. This helps us to illustrate the patterns to explain our results properly.

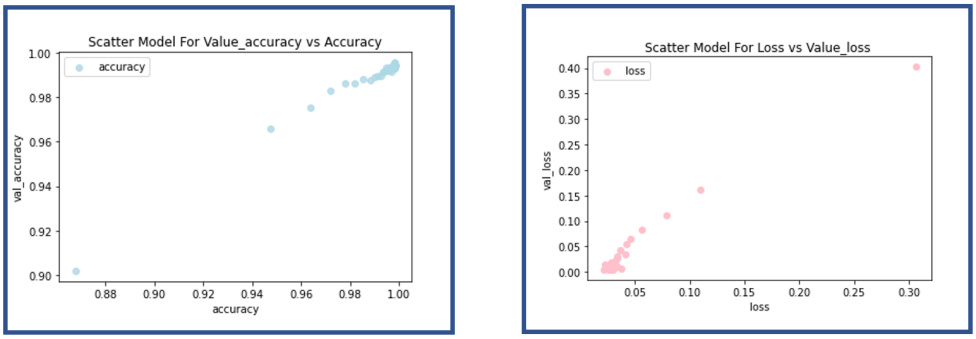


Figure 5.2: Scatter Plotting For Training and Validation

As well as scatter model figure 5.3 shows the line model for training and validation values over each epoch.

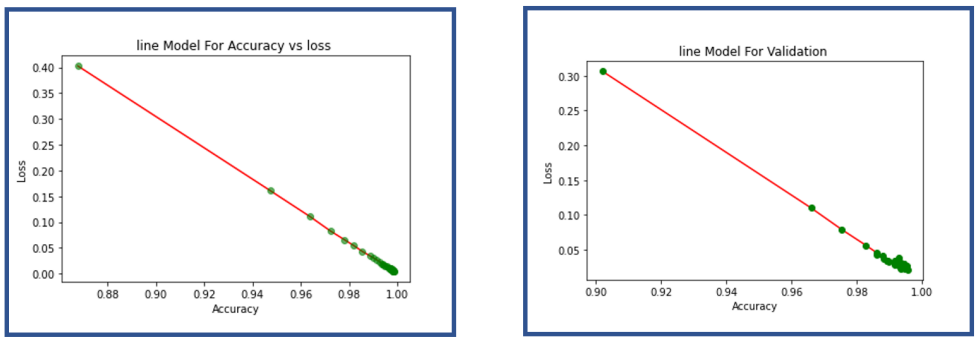


Figure 5.3: Line Plotting for Training and Validation

And finally using the prediction model we were able to predict a number of testing data with results. This helped us to generate a confusion matrix shown in figure 5.4 which can be used to further generate the accuracy, precision, recall and F1 score. The diagonal values in the matrix represent the true positive values of each class and other values represent the incorrect predictions.

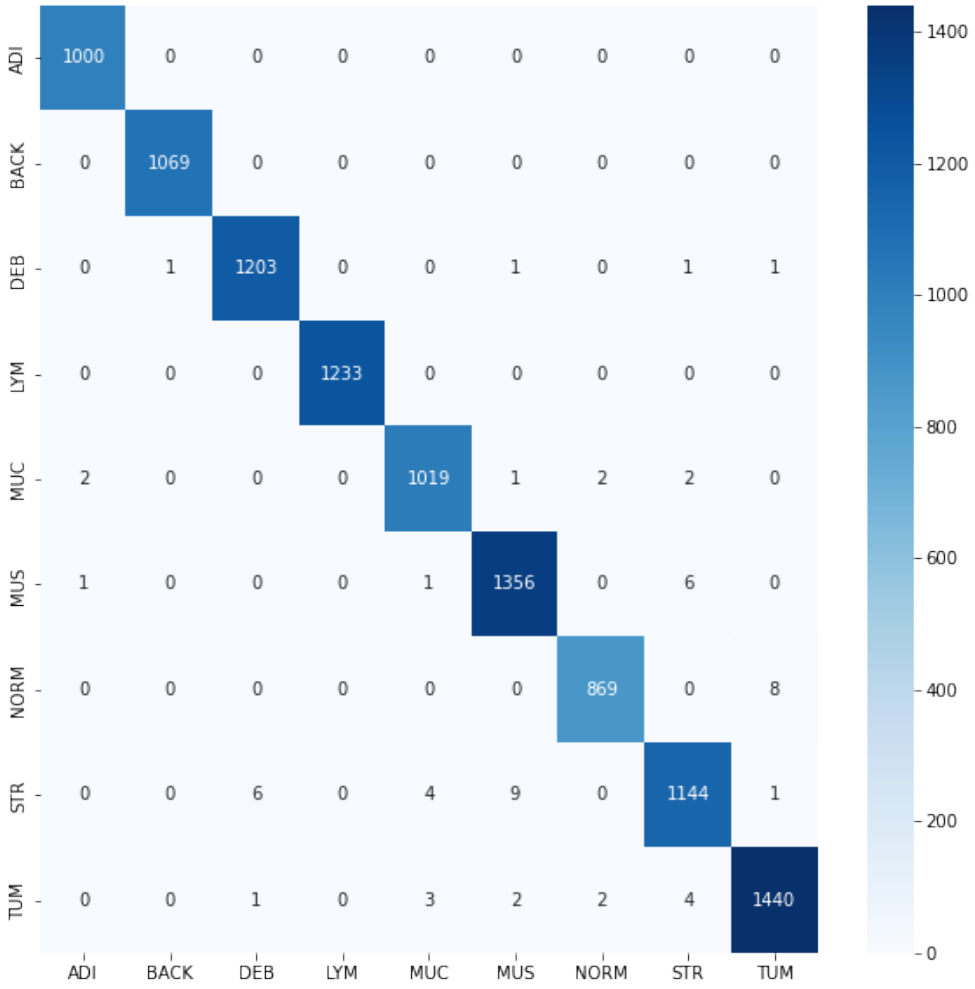


Figure 5.4: Confusion Matrix for Prediction Model

After compiling all these reports, we were able to generate the result for our ResNeXt50

model. Over 39 epochs, our experiment was able to achieve an accuracy = 0.9953, loss = 0.0262, precision = 0.9955 and recall = 0.9952. Using the formula of F1 score our f1 score is 0.9953.

$$
\begin{aligned}
F1\ Score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \\
&= 2 \times \frac{0.9955 \times 0.9952}{0.9955 + 0.9952} \\
&= 2 \times \frac{0.9907}{1.9907} \\
&= 0.9953
\end{aligned}
$$

5.2 Explainable Artificial Intelligence

We deployed XAI to visualize the patterns used for image classification after we implemented our ResNeXt50 classifier model. The output of our XAI implementation is shown in the following figure 5.5. Each class has two corresponding images. On the left image, the black box and the selection pattern is visualized. The super-pixels in the right image are colored green and red, indicating the probability of an image being allocated to a specific class. The green box in the first example (ADI) indicates super-pixels that imply to an increased probability of that image being in the ADI class. In the same example, red-marked super-pixels reduce the likelihood of the image being classified as ADI.

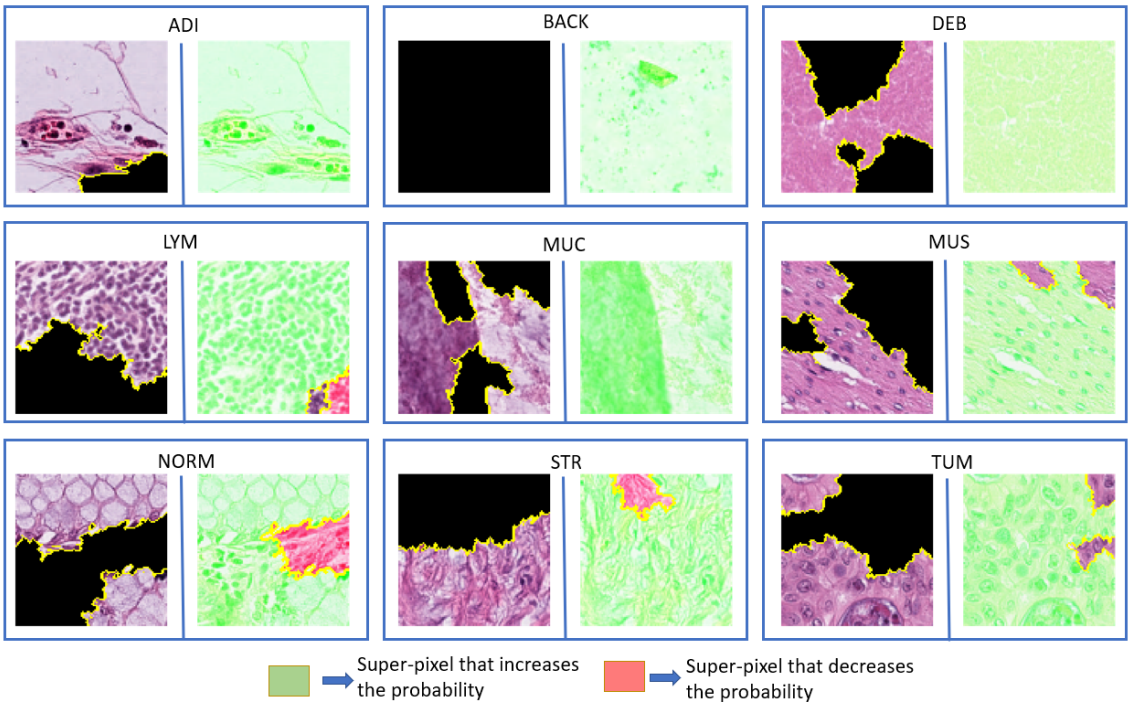


Figure 5.5: XAI Output

5.3 Comparison Between Models

We prepared a full report of accuracy, loss, and all relevant graphs after each CNN model was implemented, as indicated in the previous section. The goal was to find the ideal CNN model for implementing federated learning. Although multiple models can be used in this case, we used only the overall best model for the most efficient and keeping our research simple. Furthermore, because each model has a different execution time and epoch, the comparison between them can be improved. This step was taken on purpose to reduce overfitting in our models. As a result, depending on the total epoch run, the ultimate accuracy, loss, and F1 score may be slightly biased.

Figure 5.6 curve shows the validation accuracy and loss of each model that we have implemented over their total epoch. Each different model is labeled in different colors for differentiation.

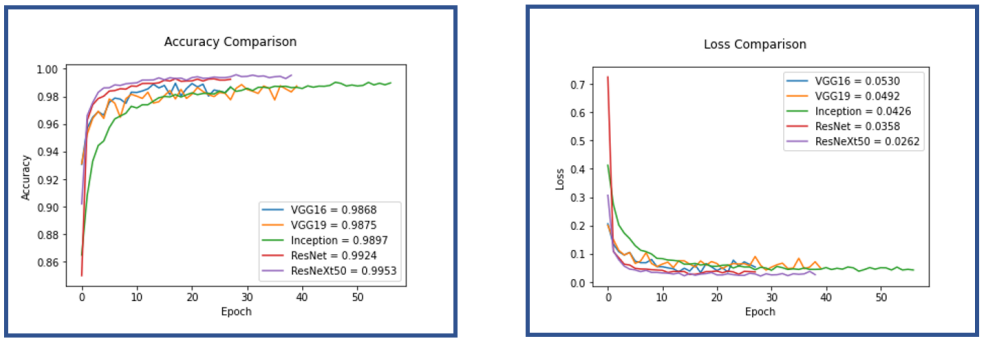


Figure 5.6: Accuracy and Loss comparison between models

Following the same formula mentioned above, precision, recall was calculated for each model. In this following table 5.1 the difference between each model is projected. For better understanding, figure 5.7 visualizes the comparison using illustrations. To differentiate between results, the chat was altered and scaled relatively. Accuracy, precision, recall and F1 score all have the same relative scaling where loss function carries a different relative scale.

Table 5.1: CNN Model Results

Models	Total Epoch	Accuracy	Loss	Precision	Recall	F1 Score
VGG16	28	0.9868	0.0530	0.9869	0.9865	0.9866
VGG19	40	0.9875	0.0492	0.9880	0.9872	0.9876
Inception v3	57	0.9897	0.0426	0.9897	0.9896	0.9896
ResNet50	28	0.9924	0.0358	0.9924	0.9923	0.9923
ResNeXt50	39	0.9953	0.0262	0.9955	0.9952	0.9953

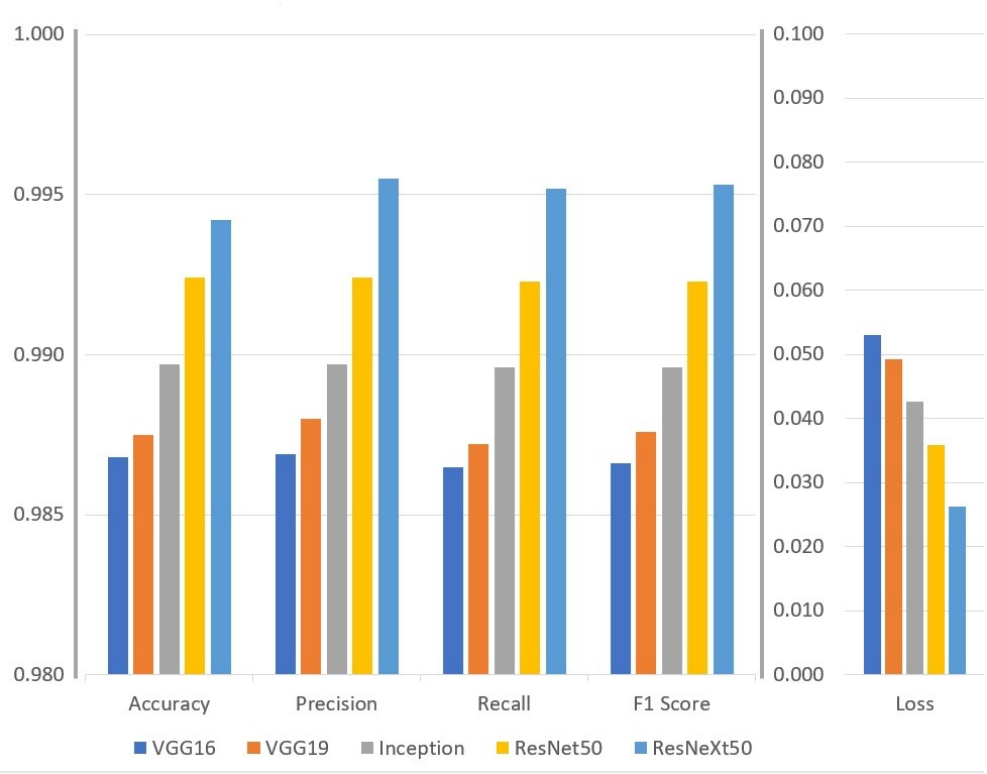


Figure 5.7: Visualization of Results in Bar Chart

Considering the results and variables, we can see that ResNeXt50 performs better in all aspects compared to VGG16, VGG19, Inception, ResNet50. For this very reason, we decided to use ResNeXt50 for the model that would be used in federated learning.

5.4 Federated Learning Results

We conferred the best CNN model (ResNeXt50) into our federated learning architecture after successfully implementing and comparing five different CNN models. Locally each client was trained first, followed by the training and updating of the returned results to the global model. This is referred to as a communication round. Our simulation included a total of 20 communication cycles. This section will discuss and visualize the results and prediction models created by our federated learning configuration, both locally and globally.

Global Communication

Figure 5.8 demonstrates how the global model was updated and optimized after each communication. The accuracy improved while the loss dropped, as we predicted in the methodology chapter. Although the rate of improvement was slow and inferior to our other CNN models, with each subsequent communication round, the final results across 20 communication rounds achieved an accuracy of 96.045%, as illustrated in the diagram below

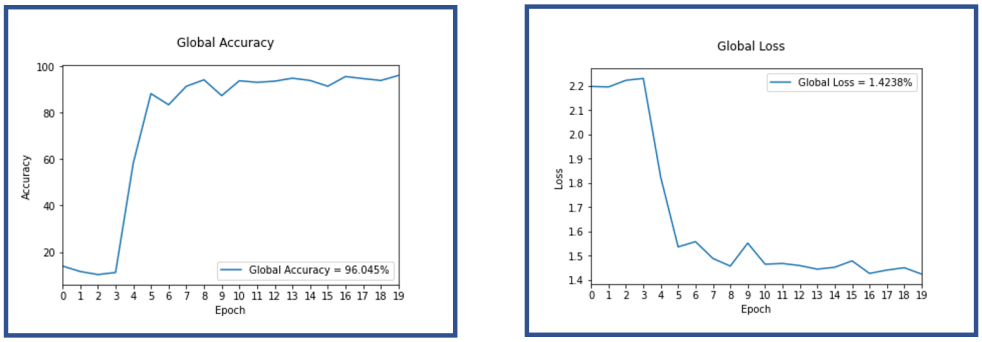


Figure 5.8: Global Accuracy and Losses in each Communication Round

Local Communication

Locally, each communication round consisted of training for all the clients. For our simulation, we decided to conduct 1 epoch for local communication rounds. After the successful simulation we were able to obtain accuracy, loss and categorical accuracy for all our clients. The accuracy for each client across 20 conversation cycles is shown in figure 5.9. Although not immediately apparent, the results show an upward tendency. That means the accuracy of the client is likely to increase, but not guaranteed.

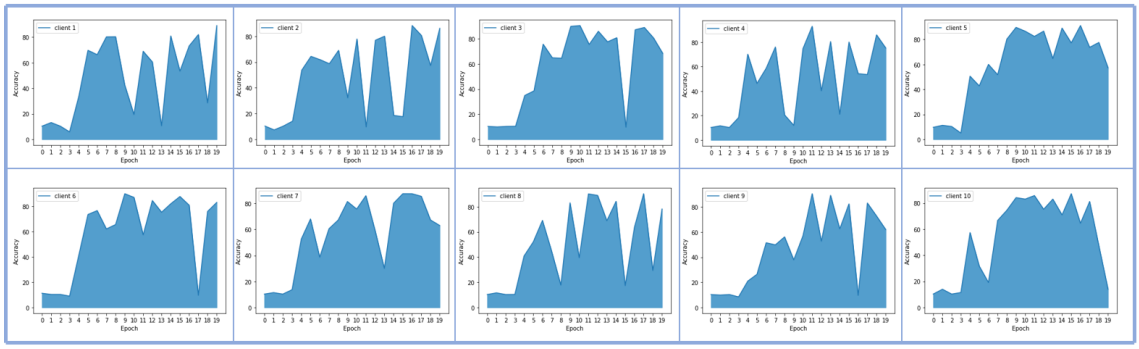


Figure 5.9: Local Accuracy For Each Clients

Furthermore, the following figure 5.10 depicts our clients loss and category accuracy across communication cycles. We can observe that, as predicted, the clients were more efficient with each subsequent communication round. The categorical accuracy shows an increasing pattern, while the loss has a decreasing pattern..

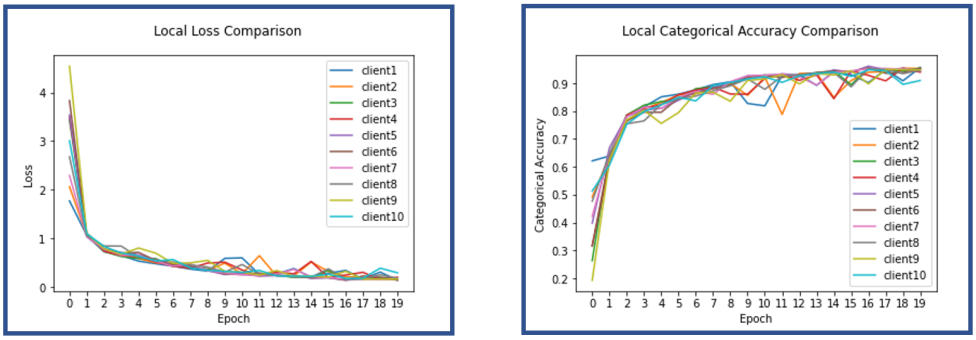


Figure 5.10: Local Loss and Categorical Accuracy Comparison of Clients

Prediction Model

After the completion of our federated learning model, we generated a prediction model to validate our findings. For that, images from the test directory were fed into the model for prediction. The generated result and prediction was then converted onto a confusion matrix which is shown in the following illustration 5.11.

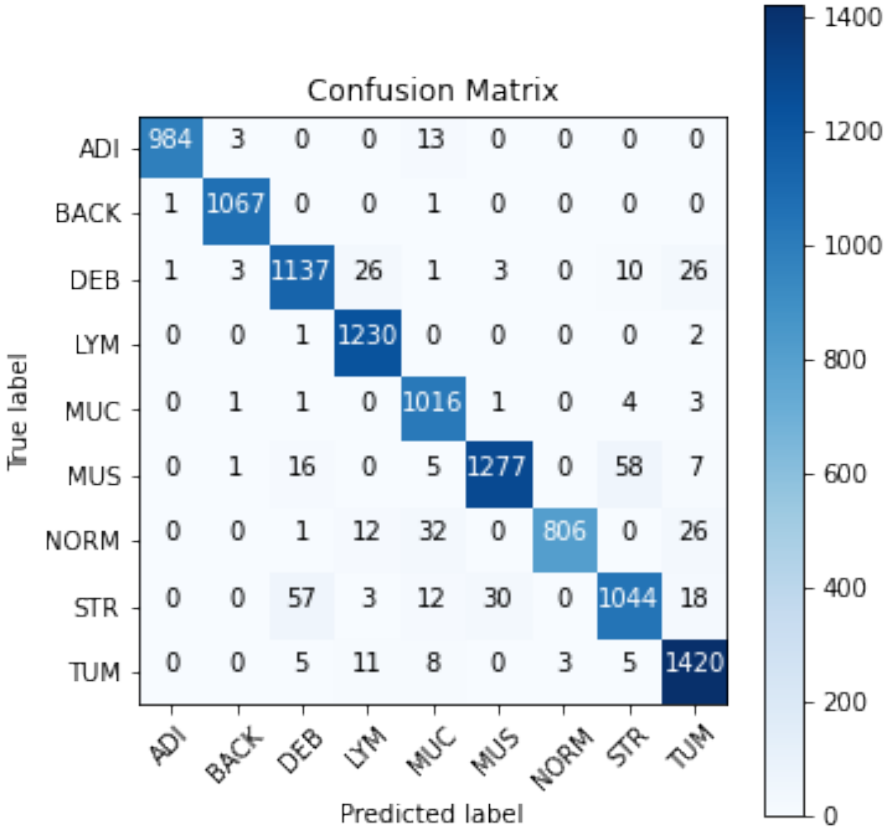


Figure 5.11: Confusion Matrix for Federated Learning Architecture

Using this, we can build a classification results table for our model through analyzing the outcome. The true positive values of each class are represented by the diagonal values in the matrix, whereas the other numbers reflect inaccurate predictions. This table 5.4 shows our model's accuracy, precision, recall and f1 score for each of our different classes. To emphasize, this table shows how well our model performs to differentiate each different class.

Table 5.2: Classification Results for FL Implementation

	Precision	**Recall**	**F1 Score**
ADI	1.00	0.98	0.99
BACK	0.99	1.00	1.00
DEB	0.93	0.94	0.94
LYM	0.96	1.00	0.98
MUC	0.93	0.99	0.96
MUS	0.97	0.94	0.95
NORM	1.00	0.92	0.96
STR	0.93	0.90	0.91
TUM	0.95	0.98	0.96
Accuracy			**0.96**
Macro Average	**0.96**	**0.96**	**0.96**
Weighted Average	**0.96**	**0.96**	**0.96**

Using the table we can plot the values of precision, recall and f1 accuracy onto a line diagram to illustrate the performance comparison of all our classes and its performance based on our model. The following bar chart, figure 5.12 displays this performance result for each class of our FL implementation.

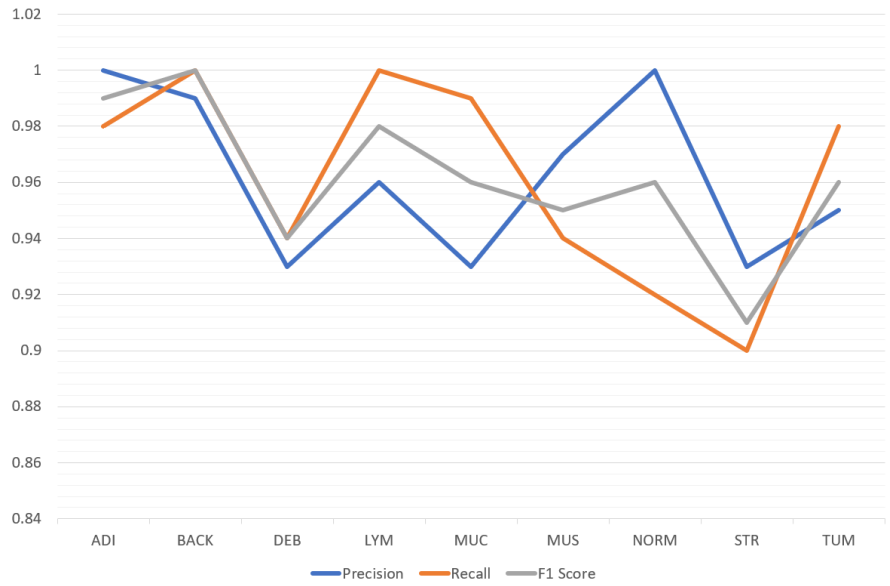


Figure 5.12: Performance of our FL Model

Chapter 6

Conclusion and Future Plan

In the sphere of medicine, digital advances in machine learning are approaching an intriguing phase. FL is a potential strategy to generate precise, stable, resilient and balanced algorithms, as all ML methodologies profit tremendously from the capacity to collect data that resembles the genuine worldwide reach. In our research we have used the distributed FL in conjunction with suitable CNN architectures to complete our categorization objective while sustaining the privacy of clients data. However, in our work the amount of cases in the training dataset for every class label was not balanced. This class imbalance problem could be solved using oversampling or undersampling techniques. Nevertheless, such research will have to wait till later. Against all odds, we have achieved a high FL accuracy of 96.045% and F1 score of 0.96. Moreover, our model was trained on a large dataset and has gone through image augmentation which prevented the chances of overfitting. Furthermore, our model is applicable in real world scenarios, for instance, it is capable of diagnosing cancer more precisely at an initial phase, reducing the number of hospitalisations or unwanted death. In spite of all the obstacles, we were able to train our model in a real world FL context, synchronized with relevant CNN models, demonstrating the potentiality of FL for developing medically applicable models without the necessity for consolidated datasets. Besides, the application of XAI has made our model accessible in time to prevent ”black box” operations. We want to further extend our framework for federated learning coalescing blockchain, focusing on the adaptability while working with fluctuating non-IID datasets with the uncertainty of throughput and computation capability.

References

[1] Debesh Jha et al. “Real-Time Polyp Detection, Localization and Segmentation in Colonoscopy Using Deep Learning”. In: *IEEE Access* 9 (2021), pp. 40496–40510. DOI: 10.1109/ACCESS.2021.3063716.

[2] Linda Rabeneck, Julianne Souchek, and Hashem B El-Serag. “Survival of colorectal cancer patients hospitalized in the Veterans Affairs Health Care System”. In: *The American journal of gastroenterology* 98.5 (2003), pp. 1186–1192.

[3] Akshay M Godkhindi and Rajaram M. Gowda. “Automated detection of polyps in CT colonography images using deep learning algorithms in colon cancer diagnosis”. In: *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*. 2017, pp. 1722–1728. DOI: 10.1109/ICECDS.2017.8389744.

[4] Marc Macenko et al. “A method for normalizing histology slides for quantitative analysis”. In: *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. 2009, pp. 1107–1110. DOI: 10.1109/ISBI.2009.5193250.

[5] Min Seob Kwak et al. “Deep convolutional neural network-based lymph node metastasis prediction for colon cancer using histopathological images”. In: *Frontiers in Oncology* 10 (2021), p. 3053.

[6] Younghak Shin et al. “Automatic Colon Polyp Detection Using Region Based Deep CNN and Post Learning Approaches”. In: *IEEE Access* 6 (2018), pp. 40950–40962. DOI: 10.1109/ACCESS.2018.2856402.

[7] Quentin Angermann et al. “Towards real-time polyp detection in colonoscopy videos: Adapting still frame-based methodologies for video sequences analysis”. In: *Computer assisted and robotic endoscopy and clinical image-based procedures*. Springer, 2017, pp. 29–41.

[8] Seung-Hwan Bae and Kuk-Jin Yoon. “Polyp Detection via Imbalanced Learning and Discriminative Feature Learning”. In: *IEEE Transactions on Medical Imaging* 34.11 (2015), pp. 2379–2393. DOI: 10.1109/TMI.2015.2434398.

[9] Nicola Rieke et al. “The future of digital health with federated learning”. en. In: *NPJ Digit. Med.* 3.1 (Sept. 2020), p. 119.

[10] Micah J. Sheller et al. “Multi-institutional Deep Learning Modeling Without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation”. In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Ed. by Alessandro Crimi et al. Cham: Springer International Publishing, 2019, pp. 92–104. ISBN: 978-3-030-11723-8.

[11] Holger R. Roth et al. “Federated Learning for Breast Density Classification: A Real-World Implementation”. In: *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*. Springer International Publishing, 2020, pp. 181–191. DOI: 10.1007/978-3-030-60548-3_18. URL: https://doi.org/10.1007%2F978-3-030-60548-3_18.

[12] Prasanna L Ponugoti, Oscar W Cummings, and Douglas K Rex. “Risk of cancer in small and diminutive colorectal polyps”. In: *Digestive and Liver Disease* 49.1 (2017), pp. 34–37.

[13] AM Leufkens et al. “Factors influencing the miss rate of polyps in a back-to-back colonoscopy study”. In: *Endoscopy* 44.05 (2012), pp. 470–475.

[14] Sharib Ali et al. “A deep learning framework for quality assessment and restoration in video endoscopy”. In: *Medical Image Analysis* 68 (2021), p. 101900.

[15] Eduardo Ribeiro, Andreas Uhl, and Michael Häfner. “Colonic polyp classification with convolutional neural networks”. In: *2016 IEEE 29th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE. 2016, pp. 253–258.

[16] Korsuk Sirinukunwattana et al. “Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images”. In: *IEEE Transactions on Medical Imaging* 35.5 (2016), pp. 1196–1206. DOI: 10.1109/TMI.2016.2525803.

[17] Meiyan Liang et al. “Identification of colon cancer using multi-scale feature fusion convolutional neural network based on shearlet transform”. In: *IEEE Access* 8 (2020), pp. 208969–208977.

[18] Jakob Nikolas Kather, Niels Halama, and Alexander Marx. *100,000 histological images of human colorectal cancer and healthy tissue*. Version v0.1. Apr. 2018. DOI: 10.5281/zenodo.1214456. URL: https://doi.org/10.5281/zenodo.1214456.

[19] Wenyuan Li et al. “Path R-CNN for prostate cancer diagnosis and Gleason grading of histological images”. en. In: *IEEE Trans. Med. Imaging* 38.4 (Apr. 2019), pp. 945–954.

[20] Christian Szegedy et al. *Rethinking the Inception Architecture for Computer Vision*. 2015. DOI: 10.48550/ARXIV.1512.00567. URL: https://arxiv.org/abs/1512.00567.

[21] Arna Fariza, Mu’arifin, and Agus Zainal Arifin. “Age Estimation System Using Deep Residual Network Classification Method”. In: *2019 International Electronics Symposium (IES)*. 2019, pp. 607–611. DOI: 10.1109/ELECSYM.2019.8901521.

[22] Zezhong Ma et al. “An Assisted Diagnosis Model for Cancer Patients Based on Federated Learning”. In: *Frontiers in Oncology* 12 (2022). ISSN: 2234-943X. DOI: 10.3389/fonc.2022.860532. URL: https://www.frontiersin.org/article/10.3389/fonc.2022.860532.

[23] Bernardo Tedeschini et al. “Decentralized Federated Learning for Healthcare Networks: A Case Study on Tumor Segmentation”. In: *IEEE Access* PP (Jan. 2022), pp. 1–1. DOI: 10.1109/ACCESS.2022.3141913.

[24] David Gunning and David Aha. “DARPA’s Explainable Artificial Intelligence (XAI) Program”. In: *AI Magazine* 40.2 (June 2019), pp. 44–58. DOI: 10.1609/aimag.v40i2.2850. URL: https://ojs.aaai.org/index.php/aimagazine/article/view/2850.