AI BASED RECTAL CANCER AND STAGE PREDICTION OVER WEB IN REAL - TIME

A PROJECT REPORT

Submitted by

BOGGADA MOUNIKA	**211418104041**
Y. EESHA SAI SRI	**211418104057**
PANCHETI DIVIJA REDDY	**211418104184**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

MAY 2022

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"AI BASED RECTAL CANCER AND STAGE PREDICTION OVER WEB IN REAL-TIME"** is the bonafide work of **"BOGGADA MOUNIKA (211418104041), Y.EESHA SAI SRI (211418104057) and PANCHETI DIVIJA REDDY (211418104184)"** who carried out the project work under my supervision.

SIGNATURE
Dr. S.MURUGAVALLI, M.E., Ph.D.,
HOD,
Department of CSE,
Panimalar Engineering College,
Chennai – 600 123.

SIGNATURE
DR. Sangeetha Karthikeyan, M.E.,
Associate Professor,
Department of CSE,
Panimalar Engineering College,
Chennai – 600 123.

Certified that the above mentioned students were examined in End Semester project viva-voice held on _____.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION BY THE STUDENT

We **BOGGADA MOUNIKA(211418104041), Y.EESHA SAI SRI(211418104057) and PANCHETI DIVIJA REDDY (211418104184)** hereby declare that this project report titled**" AI BASED RECTAL CANCER AND STAGE PREDICTION OVER WEB IN REAL-TIME",** under the guidance of **DR. SANGEETHA KARTHIKEYAN, M.E**., is the originalwork done by us and we have not plagiarized or submitted to any other degree in any university by us.

BOGGADA MOUNIKA

Y. EESHA SAI SRI

PANCHETI DIVIJA REDDY

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr. P. CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to express our heartfelt and sincere thanks to our Directors **Tmt. C. VIJAYARAJESWARI**, **Dr. C. SAKTHIKUMAR, M.E., Ph.D.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E., M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr. K. MANI, M.E., Ph.D.** histimely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr. S.MURUGAVALLI, M.E., Ph.D.,** for the support extended throughout the project.

We would like to thank my Project Guide, **Dr. SANGEETHA KARTHIKEYAN , M.E.,** andall the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

BOGGADA MOUNIKA
Y. EESHA SAI SRI
PANCHETI DIVIJA REDDY

ABSTRACT

Now-a-days, with the development of targeted therapies, many treatments are based on molecular studies, which require sampling tumor tissue from paraffin blocks for sequencing. An automated solution could potentially reduce the workload of the pathologists by acting as a screening device and may reduce the subjectivity in diagnosis. In tissue-based diagnostics, most of the work still needs to be done manually by a pathologist using a microscope to examine stained slides. The foundation of such tasks is to accurately distinguish cancer/malignant cells from normal/benign cells. However, the determination of tumor content is poorly reproducible with significant variation. As the size of tumor regions can be very small, pathologists are often required to use high magnification for detecting tumor cells. This requirement significantly increases the workload for pathologists. As digital pathology datasets have become publicly available and have opened up the possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis. In this project we introduce an application to detect Colorectalcancer based on the Convolutional Neural Network and Ranking algorithm. Here we will collect the tissue from lab or hospital and we will train the image and do data processing with segmentation and morphological filtering. Now we will store that in Azure ML server. In prediction website we will select the image and we will predict that one. The result will be displayed with ranking.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

S. NO.	ABBREVATION	EXPANSION
1	CNN	Convolutional Neural Network
2	LSTM	Long Short-Term Memory
3	MRI	Magnetic Resonance Imaging
4	MATLAB	Matrix Laboratory
5	ResNet	Residual Network
6	CRC	Colorectal Cancer

7	LGN	Lateral Geniculate Nucleus
8	SVF	Stromal Vascular Fraction
9	WAT	White Adipose Tissue
10	BAT	Brown Adipose Tissue
11	API	Application Program Interface
12	PIM	Protocol Independent Multicast
13	ROLM	Randomized On-Line Matching
14	SIANN	Space Invariant Artificial Neural
15	RNN	Recurrent Neural Network
16	EC2	Elastic Compute Cloud
17	AZ	Azure Web Services
18	AMI	Azure Machine Image
19	EBS	Elastic Block Store
20	IP	Internet Protocol
21	IPv4	Internet Protocol Version 4
22	VPC	Virtual Private Cloud

CHAPTER 1

INTRODUCTION

CHAPTER 1
INTRODUCTION

In the modern era, cancer is the most spreading complex disease. Identifying cancer without biopsy atan early stage is further imperative. Also, taking a biopsy is not good for health also. In general, cancer has been caused by hereditary instability and accumulation of multiple molecular alterations. It is also caused by cellular genes abnormal activation that controls cell growth or cell mitosis. Colorectal canceris cancer from uncontrolled cell growth in the colon or rectum. This was the third most commonly diagnosed cancer in the world. Colorectal cancer is also known as colon cancer, bowel cancer or colorectal adenocarcinoma. The main negative aspect of cancer is its diagnosis and treatment too late. Due to this problem, cancer has overtaken heart disease as the leading cause of death for any age on. Therefore, early detection of cancer is important. The images are collected and manually annotated for image processing. These images represent controlled imaging conditions and a wide variety in patient demographics. Each image has a dimension ranging from [155x240] to [960x1280] pixels with storage size of 10kB to 252kB per image.

1.1 OVERVIEW

With the development of targeted therapies, many treatments are based on molecular studies, which require sampling tumor tissue from paraffin blocks for sequencing. An automated solution could potentially reduce the workload of pathologists by acting as a screening device and may reduce the subjectivity in diagnosis. In tissue-based diagnostics, most of the work still needs to be done manuallyby a pathologist using a microscope to examine stained slides. The foundation of such tasks is to accurately distinguish cancer/malignant cells from normal/benign cells. However, the determination of tumor content is poorly reproducible with significant variation.

As the size of tumor regions can bevery small, pathologists are often required to use high magnification for detecting tumor cells. This requirement significantly increases the workload for pathologists. As digital pathology datasets have become publicly available and have opened up the possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis. In this projectwe introduce an user facing AI based application to detect and predict the stage of Rectal cancer basedon CNN with Attention mechanism and Ranking algorithm.

1.2 PROBLEM DEFINITION

Colorectal cancer is also known as colon cancer, bowel cancer or colorectal adenocarcinoma. The main negative aspect of cancer is its diagnosis and treatment too late. Due to this problem, cancer has overtaken heart disease as the leading cause of death for any age on. Therefore, early detection of cancer is important. With the development of targeted therapies, many treatments are based on molecular studies, which require sampling tumor tissue from paraffin blocks for sequencing. An automated solution could potentially reduce the workload of pathologists by acting as a screening device and may reduce the subjectivity in diagnosis. As datasets have become publicly available and have opened up the possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis. In this project we introduce an application to detect Rectal cancer based on Convolutional Neural Network and Ranking algorithm.

CHAPTER 2

LITERATURE SURVEY

CHAPTER 2
LITERATURE SURVEY

2.1- TITLE : MACHINE LEARNING FOR COLORECTAL CANCER RISK PREDICTION

AUTHOR : Ling Zheng, Elijah Eniola, Jiacun Wang

YEAR : 2021

DESCRIPTION :

Colorectal cancer is the third most prevalent cancer and the second most common cause of cancer deaths in the United States. Screening is one of the most powerful based on history of colorectal cancer and age. To facilitate a more effective screening of colorectal cancer, this paper explores the feasibility of machine learning algorithms for the colorectal cancer risk colorectal cancer risk prediction. The longitudinal Pancreatic, Lung, Colorectal, Ovarian Cancer dataset from the National Cancer Institute was utilized for the training and testing of eight machine learning algorithms. The experiment results show that the gradient boosting model has the largest area under the Receiver Operating Characteristics curve 0.82, and the random forest model has the highest accuracy 0.75, highest recall 0.76 and highest F1score 0.75. The two optimal models were also used to evaluate the importance of top risk factors, which are helpful for a more effective screening recommendation.

METHODOLOGY USED :

This paper explores the feasibility of machine learning algorithm. The machine learning algorithms are used for training and testing.

MERITS :

The two optimal methods used are top risk factors and helpful for a more effective screening recommendation.

DEMERITS:

Dataset for the training and testing requires of eight machine learning algorithms.

2.2 – TITLE : TWO STAGE CLASSIFICATION WITH CNN FOR COLORECTAL CANCER DETECTION

AUTHOR : Pallabi Sharma, Kangkana Bora, Kunio Kasugaiand Bunil Kumar

YEAR : 2020

DESCRIPTION :

In this paper, it addresses the current problem in medical image processing, the detection of colorectal cancer from colonoscopy videos. According to worldwide cancerstatistics, colorectal cancer is one of the most common cancers. The process of screeningand the removal of precancerous cells from the large intestine is a crucial task to date. The traditional manual process is dependent on the expertise of the medical practitioner.In this paper, a two-stage classification is proposed to detect colorectal cancer. In the first stage, frames of colonoscopy video are extracted and are rated as significant if it contains a polyp, and these results are then aggregated in a second stage to come to an overall decision concerning the final classification of that frame to be neoplastic and non-Neoplastic. In doing so, a comparative study is being made by considering the applicability of deep learning to perform this two-stage classification. The CNN models namely VGG16, VGG19, Inception V3, Xception, GoogLeNet, ResNet50, ResNet100, DenseNet, NASNetMobile, MobilenetV2, InceptionResNetV2 and fine-tuned version each model. It is observed that the VGG19 model is the best deep learning method for colonoscopy image diagnosis.

METHODOLOGY USED :

The CNN models namely VGG16, VGG19, inception V3,Xception, GoogleNet, resnet50, Resnet 100, densenet, NASnet mobile, mobilenetV2, inception Resnet V2 and

fine-tuned version of each model is evaluated.

MERITS :

As the process of screening and the removal of pre-cancerous cells from the large intestine is a crucial task by using these, we can reduce the task.

DEMERITS:

It fine-tuned version of each model is evaluated.

2.3 – TITLE : GRADING OF COLORECTAL CANCER USING HISTOLOGY IMAGES.

AUTHOR : Namita Sengar; Neeraj Mishra, Malay Kishore Dutta, Jiri Prinosil, Radim Burget

YEAR : 2020

DESCRIPTION :

This paper proposed an automated system for grading of colorectal cancer using image processing methods. Almost half a million people die every year due to coloncancer. Histopathological tissue analysis is a common method for its detection, which needs an expert pathologist. Screening for this cancer is effective for prevention as well as early detection. The method proposed segment the glands automatically by using intensity-based thresholding and organizational properties for classification. In existing literature, the majority of studies are based on gland segmentation in healthy or benign samples, but rarely on intermediate or high grade cancer. Unlike most of the existing methods this system is fully automated and grades the images as benign healthy, benign adenomatous, moderately differentiated malignant and poorly differentiated malignant. The proposed method achieves overall accuracy of 81% when tested on 165 histology images.

METHODOLOGY USED :

Image processing methods.

MERITS:

We can predict at its earlier stag**e.**

DEMERITS:

It gives less accuracy result and its long process.

2.4 – TITLE : AUTOMATIC CLASSIFICATION OF NON-INFORMATIVE FRAME IN COLONOSCOPY VIDEOS

AUTHOR : Ballesteros, Trujillo and C. Mazo

YEAR : 2020

DESCRIPTION:

Colonoscopy is the most recommended test for prevention of colorectal cancer. Nowadays, digital videos are recorded during colonoscopy procedures and used for training machine learning algorithms. Machine learning algorithms are used for automatically recognizing lesions based on supervised learning. Moreover, annotation of lesions is a difficult and time consuming process that is manually made by gastroenterologists. Those annotations may contain frames that have not useful information, called non-Informative frames. The presence of non-Informative frames in a group of frames labelled as lesion affects the accuracy of machine learning algorithms. In this paper, a method based on edge detection is proposed to automatically classify a frame -from a colonoscopy video - into either Informative and Non-Informative. Non-Information Frames usually do not contain many edges. However, brightness regions produce false edges. Therefore, the proposed method includes a technique for brightness segmentation to identify false edges. The proposed method is evaluated using videos

annotated by gastroenterologists. Elimination of No - Informative frames may reduce significantly the number of frames to be annotated by gastroenterologists and may improve the accuracy of machine learning algorithms. Experimental evaluation showed that the accuracy and the precision of the proposed method is over 95%.

METHODOLOGY USED:

A random forest classifier was used for classification. An enhanced edge detection-based method was proposed.

MERITS:

It includes a technique for brightness segmentation to have accurate false edges.

DEMERITS:

1. Presence of Non-Information frames in a group of frames labelled as lesion affects the accuracy of machine learning algorithms.
2. Time consuming process.

2.5 – TITLE : NON-INFORMATIVE FRAME CLASSIFICATION IN COLONOSCOPY VIDEOS USING CNN

AUTHOR : A. B. M. R. Isla, A. Alammari, W. Tavanapong, J. wong and P. C.de groen.

YEAR : 2019

DESCRITION :

In the US, colorectal cancer is the second leading cause of cancer-related deaths behind lung cancer, causing about 49,000 annual deaths. Colonoscopy is currently the gold standard procedure for colorectal cancer screening. However, recent data suggest that there is a significant (4-12%) miss-Rate for the detection of even large polyps and

cancers. To address this, we have been investigating an 'automated feedback system' which measures quality of colonoscopy automatically by analyzing colonoscopy video frames in order to assist the endoscopist to improve the quality of the actual procedure being performed. One of the fundamental steps analyzing colonoscopy video frames for the automated quality feedback system is to distinguish non-informative frames from informative ones. Most methods to detect and classify these non-informative frames are based on the hand-engineered features. However, it is very tedious to design optimal hand-engineered features. In this paper, we explore the effectiveness of Convolutional Neural Network (CNN) to detect and classify these non-informative frames. The experimental results show that the proposed approaches are promising.

METHODOLOGY USED:

A CNN model was used with random trained dataset.

MERITS:

Easy for implementation and reduces the hand engineered work.

DEMERITS:

It is very tedious to design optimal hand-engineered features.

CHAPTER 3
SYSTEM ANALYSIS

CHAPTER-3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In the existing system, the concept of data mining systems supported on cancer prophecy system merging the prediction scheme with mining tools. The categorization algorithms used in the existing system is called decision tree. The user enters into the cancer prophecy scheme, and then required to retort the queries, connected to genetic and non-genetic skin textures. In that case the prediction structure allots the hazard rate to both query bases on the client retorts. One time the exposure significance is estimated, the series of the coercion preserve is resolute by the forecast structure. Research data shows that the accuracy of cancer prediction system is about 73%.

3.2 PROPOSED SYSTEM

With the development of targeted therapies, many treatments are based on molecular studies, which require sampling tumor tissue from paraffin blocks for sequencing. An automated solution could potentially reduce the workload of pathologists by acting as a screening device and may reduce the subjectivity in diagnosis. In tissue-based diagnostics, most of the work still needs to be done manuallyby a pathologist using a microscope to examine stained slides. The foundation of such tasks is to accurately distinguish cancer/malignant cells from normal/benign cells. However, the determination oftumor content is poorly reproducible with significant variation. As the size of tumor regions can bevery small, pathologists are often required to use high magnification for detecting tumor cells. This requirement significantly increases the workload for pathologists. As digital pathology datasets have become publicly available and have opened up the possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis.

As digital pathology datasets have become publicly available and have opened up the possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis. In this project we introduce an user facing AI based application to detect and predict the stage of Rectal cancer based on CNN with Attention mechanism and Ranking algorithm.

3.3 FEASIBILITY STUDY

Reduce the workload of pathologists by acting as a screening device and also reduce the subjectivity in diagnosis.

Inclusion of feature in Scanning Devices for Quick analysis.

Possibility of evaluating the feasibility of applying deep learning techniques to improving the efficiency and quality of histologic diagnosis.

3.3.1 AREAS OF FEASIBILITY

Economic Feasibility:

The financial cost related to this project it feasible as it only requires trained Model and system with good processing power.

- Total number of lines of code(LOC) = 2000K
- KLOC = 2000/1000 = 2
- Effort = 2.4*(2)^1.05 = 4.969 person-month
- Development time = 2.5(4.969)^0.38 = 4.597 months
- Average staff size = 4.969/4.597 = 1.0809 person
- Productivity = 2/4.969 = 0.402 KLOC/person-month
- P = 402 LOC/person-month

Hence, it’s clear that this project is economically feasible

Technical Feasibility:

- It is related to the feasibility of training the model and implementing it in an web application.
- Since the system implementation relies on processing power a decent machine with good processingcapability is required.
- This project is based on machine learning algorithms and the technologies are :
    1. Machine learning algorithm – CNN
    2. Artificial Intelligence
    3. Azure ML
    4. IDE : visual studio

Schedule Feasibility:

Based on the designed timeline chart the proposed system only requires 2-3 months for developingit without any delay.

epoch	train_loss	valid_loss	accuracy	time
0	0.839425	0.354545	0.881119	12:13
1	0.521362	0.425943	0.867133	12:15
2	0.478811	0.337351	0.899101	12:04
3	0.377444	0.358566	0.884116	12:06
4	0.328109	0.252855	0.915085	12:16
5	0.266330	0.210688	0.937063	12:05
6	0.208004	0.174468	0.946054	12:14
7	0.175398	0.172691	0.946054	12:14

Fig. No. 3.1 Training with Epoch

Using 7 epochs we are getting is 94%accuracy.

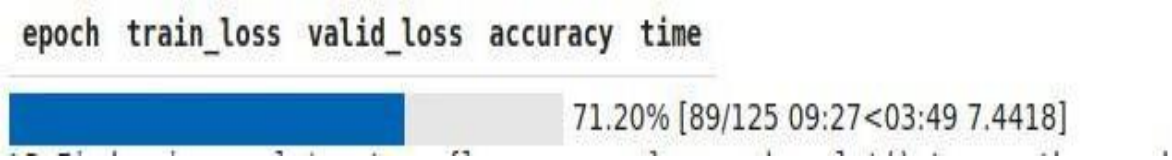


Fig. No. 3.2 Accuracy with Epoch

3.4 HARDWARE ENVIRONMENT

Processor	-	Pentium –IV
Speed	-	1.1 Ghz
RAM	-	4GB RAM
Hard Disk	-	20 GB
Key Board	-	Standard Windows
Mouse	-	Two or Three Button Mouse
Monitor	-	ANY

3.5 SOFTWARE ENVIRONMENT

Operating System	-	Unix/Linux/XP/7/8/8.1/10
Coding Language	-	Python >= 3.8.0
		Flask

CHAPTER 4

SYSTEM DESIGN

CHAPTER 4
SYSTEM DESIGN

4.1 ER Diagram

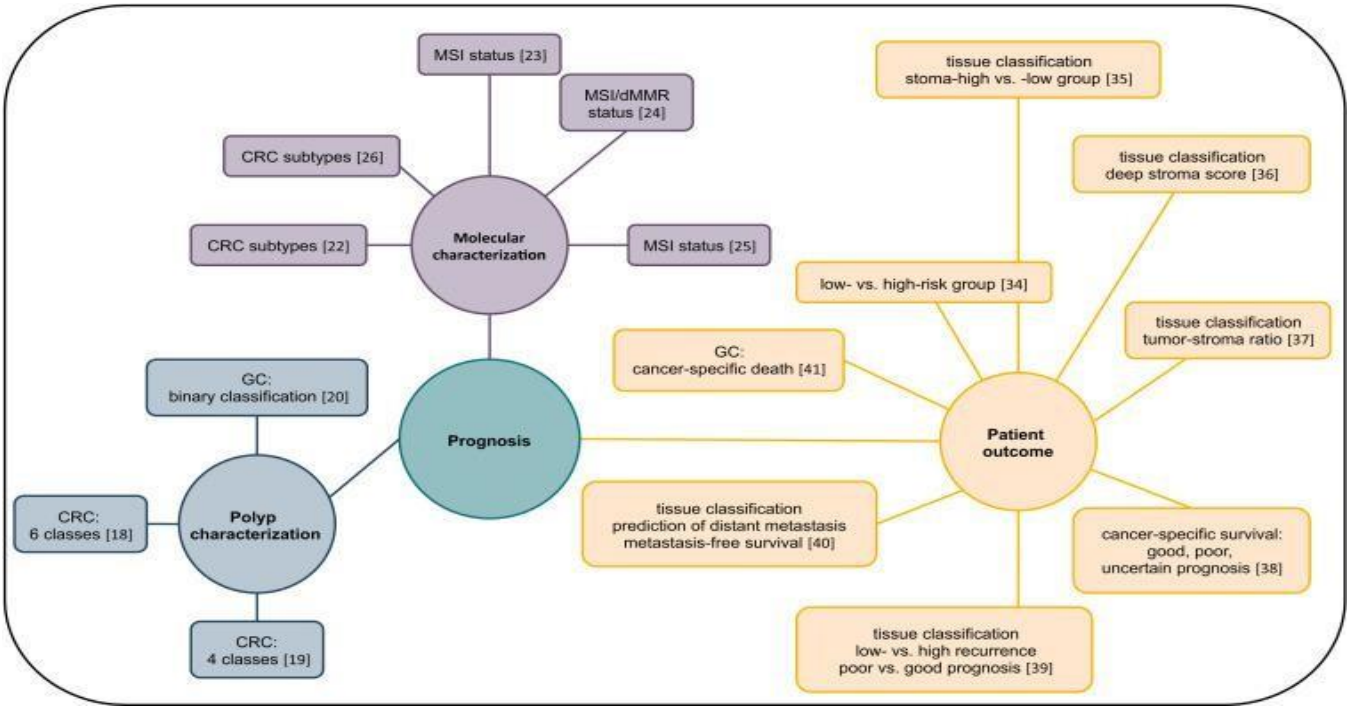


Fig. No. 4.1 ER Diagram

In the above diagram, the Prognosis connects with patient outcome, Molecular characterization and Polyp characterization.

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.

4.2 DATA DICTIONARY

The diagram depicts the data dictionary of the project. The entity ‘pixel’ has the attribute image, numpyis the data type. The entity ‘dataset’ has the attribute class with string as data type.

ENTITY	ATTRIBUTE	DATA TYPE	DESCRIPTION
pixel	Image	numpy.array	input image
dataset	Class	string	dataset label
preprocess	size	int	image size
augmentation	angle	float	rotation range
annotation	annotated file	json	annotated cancer part in image
trained model	model	h5	saving model

Table No.4.1 Data Dictionary

The entity 'pre-processor' has the attribute size and integer as data type. The entity 'augmentation' has angle as attribute and float as data type. The entity 'annotation' has annotated file as attribute and json as data type. The final entity 'trained model' has attribute model and h5 as data type.

4.3 DATABASE DIAGRAM

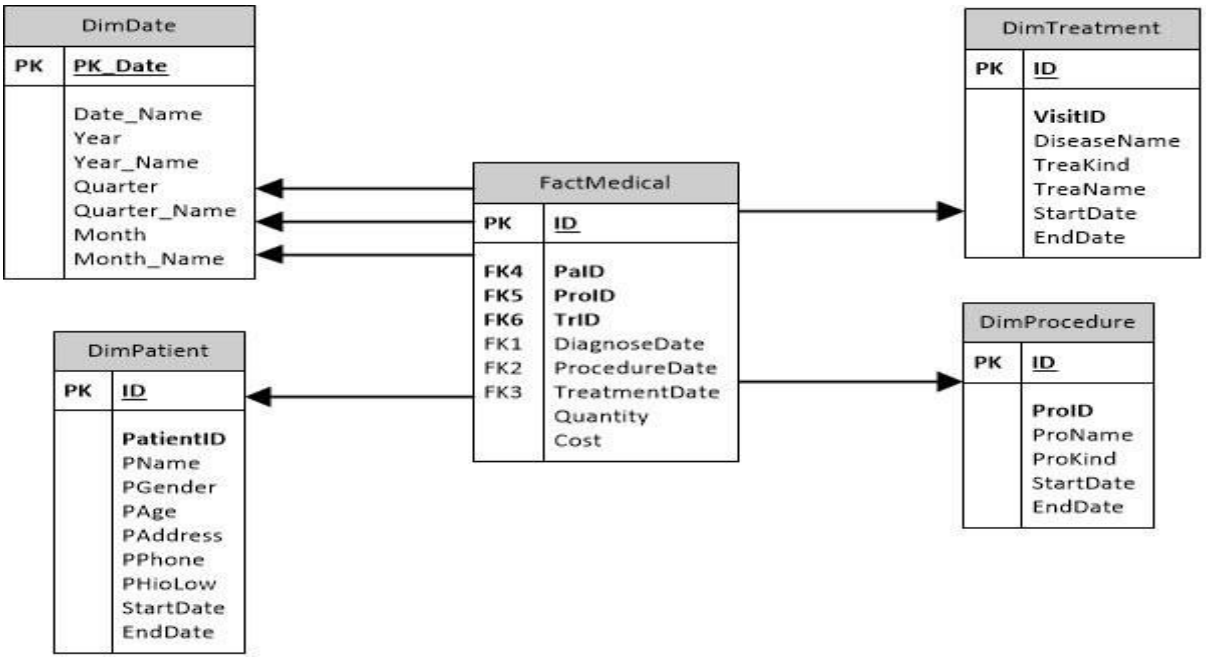


Fig. No. 4.2 Database Diagram

In the above diagram, the class is static diagram and it is used to model the static view of the system. The static view describes the vocabulary of the system.

4.4 DATA FLOW DIAGRAM

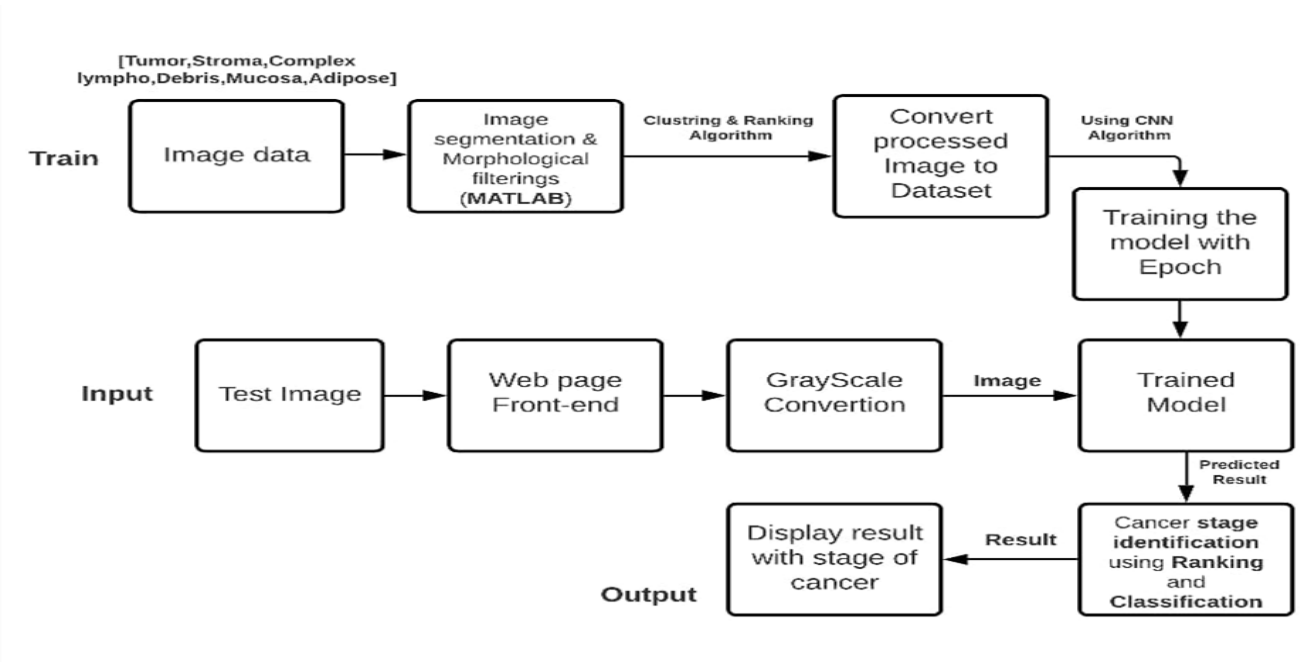


Fig. No.4.3 Data flow Diagram

- In training phase, we collect tissue datasets and apply morphological filtering's.
- Second stage is clustering and ranking algorithm
- Clustering is used to group the images that comes under tumor, stroma etc., separately
- Ranking is used to rank the image
- Then we train the dataset using CNN algorithm, we train using epoch

4.5 UML DIAGRAM

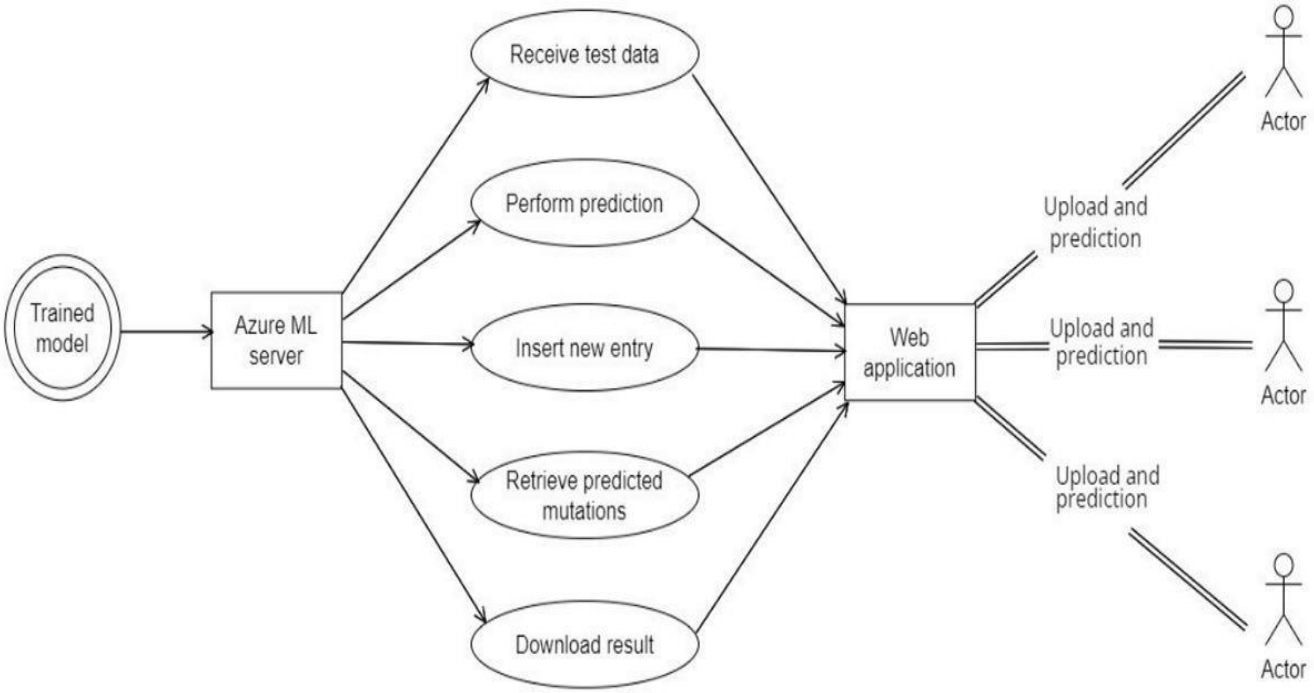


Fig. No. 4.4 UML Diagram

- In the above diagram, the images are trained and we get the trained model.
- The trained model is stored in Azure ML server.
- Though that we will perform prediction, insert new entry, retrieve predicted mutations and download result in web application.
- Through the web application, we will receive test data, then perform prediction, insert new entry, retrieve predicted mutual and download result in web application. In web application we will upload and predict then we will send to actor .

4.6 UI DIAGRAM

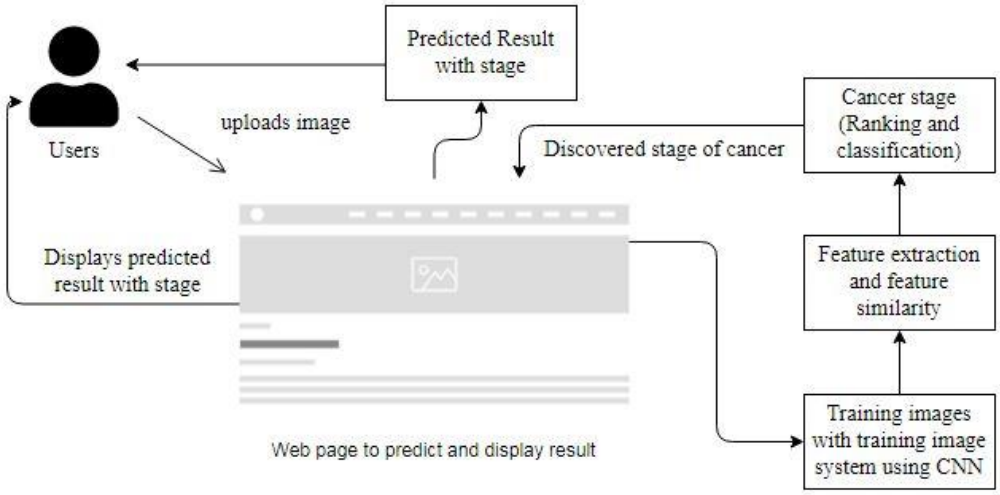


Fig. No.4.5 UI Diagram

- Here, the user uploads image in the webpage for prediction.
- The uploaded images are trained using CNN algorithm and the cancer stage is predicted.
- The predicted result with stage is displayed to the user in the webpage.

CHAPTER 5

SYSTEM ARCHITECTURE

CHAPTER 5
SYSTEM ARCHITECTURE

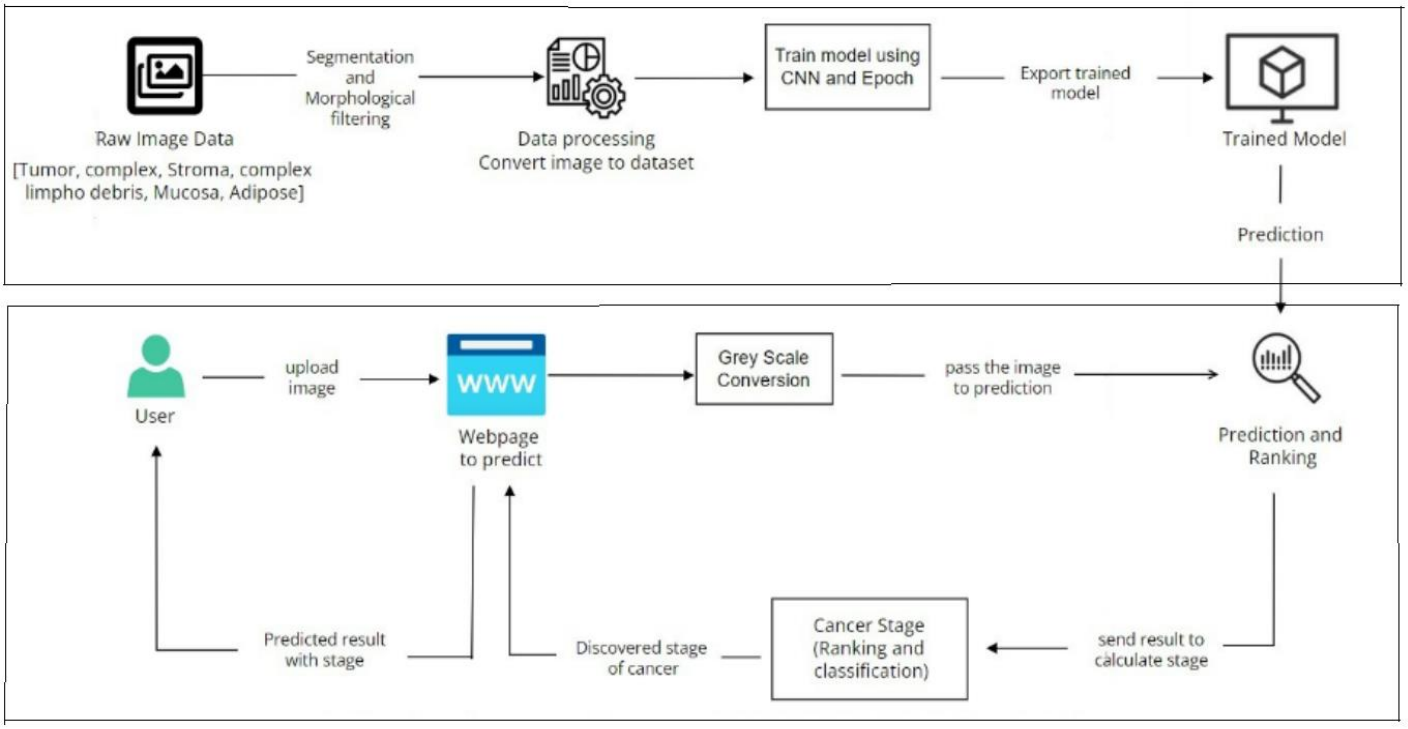


Fig. No. 5.1 Architecture Diagram

5.1 MODULE DESIGN SPECIFICATION

There are 4 modules:

1.Dataset preparation and pre-processing.

2.Dataset splitting.

3.Modelling.

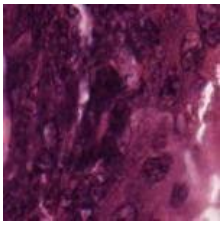
4.Model deployment over web.

1. **Dataset preparation and pre-processing:**

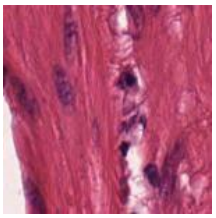
Data is the foundation for any machine learningproject. The second stage of project implementation is complex and involves data collection, selection, preprocessing, and transformation which include:

i) Data collection:

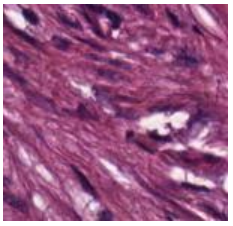
The image is collected from the external source via endoscopic ultrasound which uses ultrasound imaging and endoscopy to determine abnormalities in the colon. A special endoscope uses high-frequency sound waves to produce detailed images of the lining and walls of your digestive tract and chest. We have collected around 5000 images i.e., under each tissue there are 645 images. Parameters Of the image such as brightness, contrast and exposure are maintained such that the minor variability in them does not cause any deviation in the final output. The images are examined individually as it may contain unwanted noise which may be removed using image segmentation and morphological filtering.



Tumor



Stroma



Complex

ii) Data preprocessing:

Here the collected tissues are given as raw data set to train the model for the prediction. After image conversion the images undergo image segmentation.

Image segmentation involves converting an image into a collection of regions of pixels that are represented by a mask or a labeled image. By dividing an image into segments, you can process only the important segments of the image instead of processing the entire image.

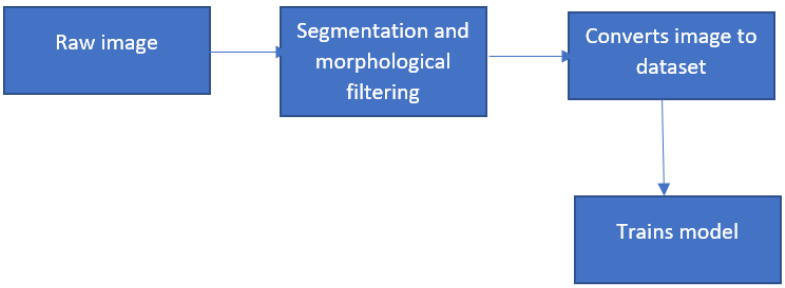


Fig. No. 5.2 Data Processing

iii) Data transformation:

After training the data user has to select the image for the prediction using CNN algorithm.

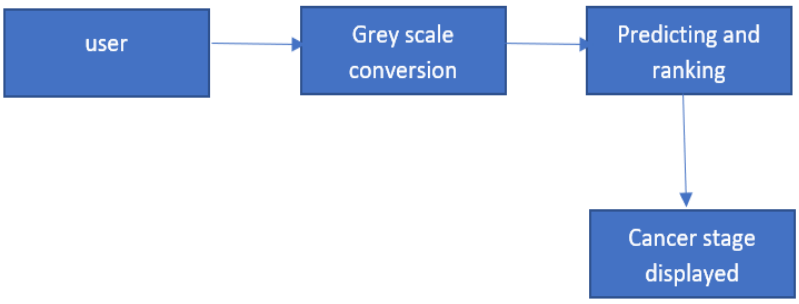


Fig. No. 5.3 Data Transformation

iv) Data visualization:

Here the data is represented the chart as we split the types of cancer in the form of distributed data.

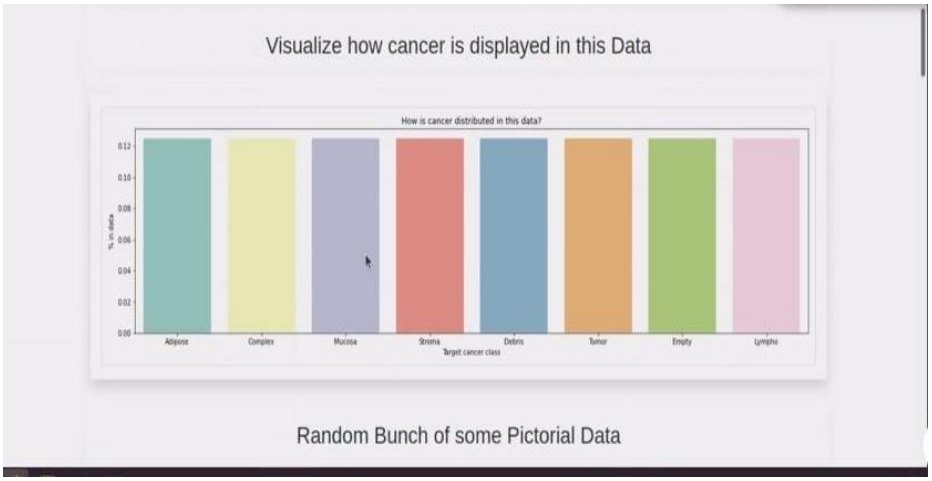


Fig. No. 5.4 Data Visualization

2. Data splitting:

i) Training set:

A data scientist uses a training set to train a model and define it optimal parameters it has to learn from data.

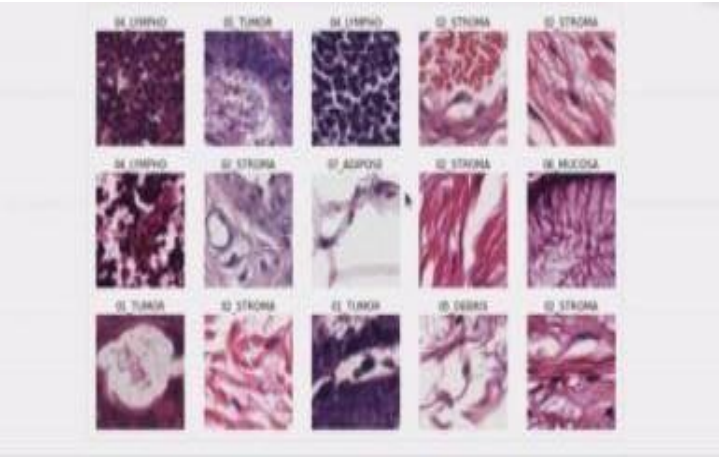


Fig. No. 5.5 Training set

ii) Test set:

A test set is needed for an evaluation of the trained model and its capability for generalization.

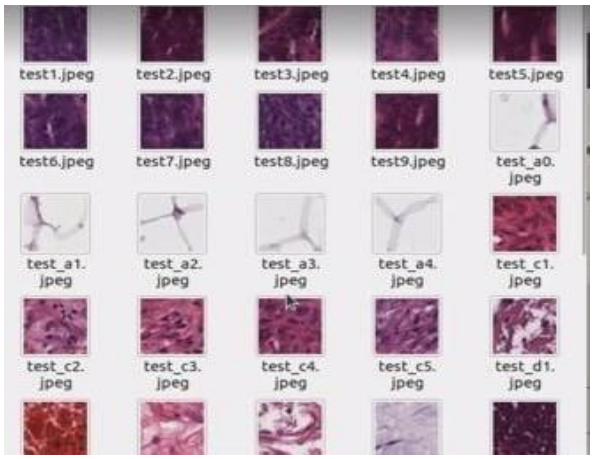


Fig. No. 5.6 Test set

3. Modelling:

During this stage we train numerous models to define which one of them provides the most accurate prediction.

i) Model training:

Here the human tissues collected from the lab/hospitals are to be drained and converted to datasets.

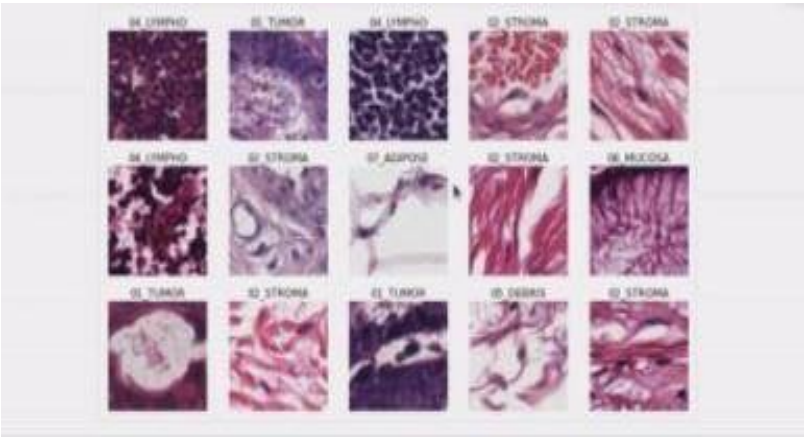


Fig. No. 5.7 Model Training

ii) Model evaluation and testing:

This process is done by the user as the trained images are available and need to be selected and the stage of the cancer is predicted. This is done in the web page.

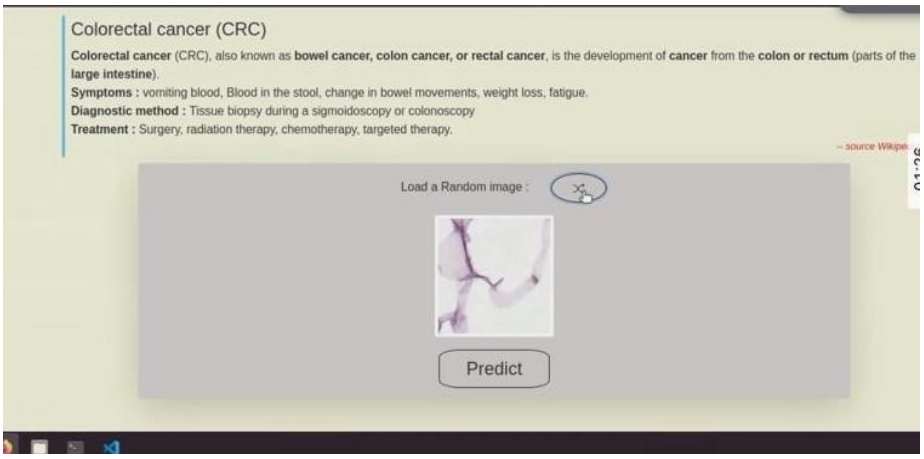


Fig. No. 5.8 Model evaluation and testing

4. Model deployment over web:

After selecting the image by the result as the cancer prediction and ranking is done over the web.

5.2. ALGORITHMS

5.2.1 CLUSTERING

Cluster analysis or clustering is the task of grouping a set of objects in sucha way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their understanding of what constitutes a cluster and how to efficiently find them.

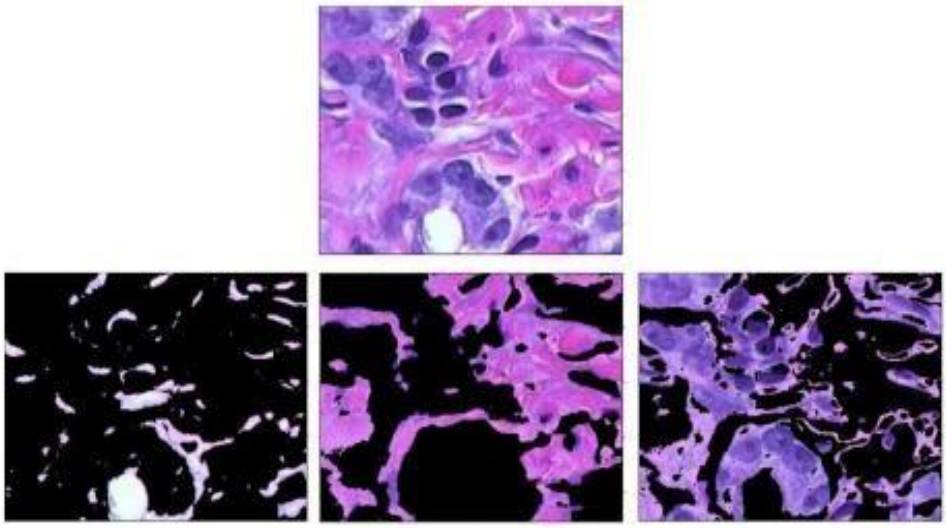


Fig. No. 5.9 Image Segmentation using Clustering

- It is a main task of exploratory data analysis, and a common technique image
- analysis and machine learning.
- The dataset contains a number of images. Clustering is used to group the images
- in the dataset under tumor, stroma etc.
- It is necessary to modify data preprocessing and model parameters until the result achieves the desired properties.

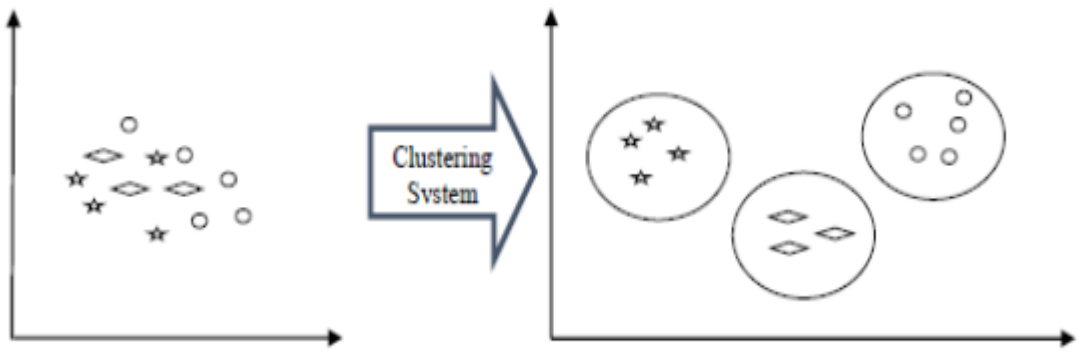


Fig. No. 5.10 Clustering of objects

5.2.2 RANKING ALGORITHM

Randomized On-Line Matching, a representative of a class of algorithms, isa sequential algorithm that exploits a randomized efficient on-line matching algorithm that calculates maximal matchings in bipartite graphs, named theRanking algorithm, as its basis. The Ranking algorithm makes a matching decision considering one output after the other. Specifically, during every switch cycle, theRanking algorithm calculates the (maximal) matching, incrementally with the following steps:

S1: Calculate a random permutation $\pi(In)$ (ordering) of inputs, which is thesame for all outputs

S2: Consider outputOut[0] and identify the requests to it (i.e., the first inputin $\pi(In)$ that has a request for Out[0]; the requests of the selected input are deleted from the graph)

S3: MatchOut[0] to the eligible input (if any) of highest rankS4:

Repeat Steps S2 and S3 for all remaining outputs.

```
40
test1.jpeg      [1.933792e-42 5.986377e-29 1.363851e-26 2.350363e-35 1.000000e+00 1.740061e-31 1.103803e-41 0.000000e+00]
test2.jpeg      [4.028733e-42 4.910942e-36 7.982636e-26 1.129785e-31 1.000000e+00 3.002878e-30 8.925758e-39 0.000000e+00]
test3.jpeg      [7.006492e-45 1.082017e-30 1.014074e-25 7.141852e-39 1.000000e+00 2.456510e-32 0.000000e+00 0.000000e+00]
test4.jpeg      [1.401298e-45 8.426362e-33 1.007098e-30 6.524866e-40 1.000000e+00 2.232796e-35 0.000000e+00 0.000000e+00]
test5.jpeg      [1.196026e-38 9.537043e-24 5.638265e-22 3.823390e-29 1.000000e+00 2.819118e-27 2.495867e-35 1.205957e-41]
test6.jpeg      [1.821688e-44 2.112725e-37 3.073386e-30 1.874856e-39 1.000000e+00 4.324794e-35 0.000000e+00 0.000000e+00]
test7.jpeg      [1.482181e-40 4.235194e-28 4.214252e-24 2.620782e-33 1.000000e+00 4.212170e-28 5.004233e-40 0.000000e+00]
test8.jpeg      [9.809089e-44 2.237903e-28 2.273661e-24 3.345538e-35 1.000000e+00 1.851981e-32 2.569986e-39 0.000000e+00]
test9.jpeg      [2.522337e-44 1.071986e-26 7.159664e-24 5.136255e-33 1.000000e+00 1.527526e-29 8.427409e-42 0.000000e+00]
test_s1.jpeg    [0.000000e+00 1.056101e-38 4.397695e-41 0.000000e+00 1.000000e+00 1.008935e-43 0.000000e+00 0.000000e+00]
test_s2.jpeg    [0.00000e+00 3.16656e-40 1.41391e-42 0.00000e+00 1.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00]
test_s3.jpeg    [0.0000e+00 4.3231e-41 1.2612e-44 0.0000e+00 1.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00]
test_s4.jpeg    [0.000000e+00 8.091807e-35 2.493477e-36 1.583467e-43 1.000000e+00 2.147205e-39 0.000000e+00 0.000000e+00]
```

Fig. No. 5.11 Output for Ranking Algorithm

There are seven different types of cancer cells are as follows:

· Tumor

· Stroma

· Complex

· Lymph

· Debris

· Mucosa

· Adipose

The ranking algorithm is used to rank the images.The above-mentioned tissues are allotted a rank number each and the images in the dataset is ranked based on their group.
For example, tumor – rank no 1, stroma – rank no 2, and so on.

5.2.3. CNN ALGORITHM

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on the shared-weight architecture of the convolution kernels that shift over input features and provide translation equivariant responses. Counter-intuitively, most convolutional neural networks are only equivariant, as opposed to invariant, to translation. They have applications in image and video recognition, recommender systems, image classification, Image segmentation, medical image analysis, natural language processing, brain-computer interfaces, and financial time series are regularized versions of multilayer perceptron.

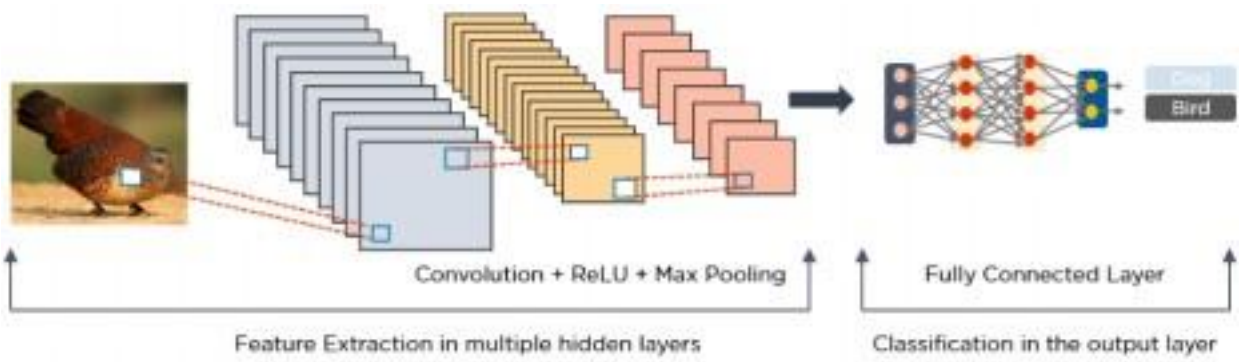


Fig. No. 5.12 Image processed via CNN

Multilayer perceptron usually means fully connected networks, that is, each neuron in one layer isconnected to all neurons in the next layer. The "full connectivity" of these networks makes them prone to overfitting data. Typical ways of regularization, or preventing overfitting, include: penalizing parameters during training (such as weight decay) or trimming connectivity. CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using smaller and simpler patterns embossed in their filters. Therefore, on a scale of connectivity and complexity, CNNs are on the lower extreme.

CHAPTER 6

SYSTEM INPLEMENTATION

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1. TRAINING CNN AND LSTM

```
from fastai import *
from fastai.vision import *
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import auc,roc_curve
import os
print (os.listdir("/content/drive/My Drive/Colab Notebooks"))
% Matplotlib inline
class_names = {1: "Tumor", 2: "Stroma", 3: "Complex", 4: "Lympho",
            5: "Debris", 6: "Mucosa", 7: "Adipose", 8: "Empty"}
class_numbers = {"Tumor": 1, "Stroma": 2, "Complex": 3, "Lympho": 4,
            "Debris": 5, "Mucosa": 6, "Adipose": 7, "Empty": 8}
class_colors = {1: "Red", 2: "Orange", 3: "Gold", 4: "Limegreen",
             5: "Mediumseagreen", 6: "Darkturquoise", 7: "Steelblue", 8: "Purple"}
label_percentage = df.label.value_counts() / df.shape[0]
class_index = [class_names[idx] for idx in label_percentage.index.values]
plt.figure(figsize=(20,5))
sns.barplot(x=class_index, y=label_percentage.values, palette="Set3");
plt.ylabel("% in data");
plt.xlabel("Target cancer class");
plt.title("How is cancer distributed in this data?");
tfms=get_transforms(flip_vert=True, max_warp=0.)
```

```
tfms=get_transforms(flip_vert=True, max_warp=0.)
data = (ImageList.from_folder(path)
        .split_by_rand_pct()
        .label_from_folder()
 .transform(tfms, size=150)
 .databunch(num_workers=2, bs=32))
learner= cnn_learner(data, models.resnet34, metrics=[accuracy],
model_dir='/content/drive/My Drive/Colab Notebooks')
# Train the model on 4 epochs of data at the default learning rate
#learner.fit_one_cycle(4)
## Fit the model over 8 epochs
lr=5e-3  ## uncomment this
learner.fit_one_cycle(8, lr)   ## uncomment this
#save the model
learner.save('/content/drive/My Drive/Colab Notebooks/level-1')
#print(os.listdir("./drive/My Drive/Colab Notebooks"))
#load the model
#learner.load('level-1')
#save the model
learner.save('level-2') ## uncomment this
#load the model
#learner.load('level-1')
# intrepting most confused
interp.most_confused()
#  ROC curve
fpr, tpr, thresholds = roc_curve(lb.numpy(), preds.numpy()[:,1], pos_label=1)
#  ROC area
pred_score = auc(fpr, tpr)
print(f'ROC area is {pred_score}')
```

```
plt.figure()
plt.plot(fpr, tpr, color='green', label='ROC curve (area = %0.2f)' % pred_score)
plt.plot([0, 1], [0, 1], color='red', linestyle='--')
plt.xlim([-0.01, 1.0])
plt.ylim([0.0, 1.01])
plt.xlabel('False_Positive_Rate')
plt.ylabel('True_Positive_Rate')
plt.title('Receiver_Operating_Characteristic')
plt.legend(loc="lower right")
####/**************Testing and prediction (load level-2)**************/###
#learner.load("level-2")
learner.load("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/old_level-1")
####/**************Testing and prediction **************/###
#learner.load("level-2")
# lets save our model with two formats: pkl and pth
#learner.export('pkl_colorectal_CNN_model.pkl')
#learner.save('pth_colorectal_CNN_model')
imageC1=random.choice(os.listdir("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/04_LYMPHO/"))
#read = cv2.imread("/content/drive/My Drive/Colab Notebooks/test.jpeg")
#test_image=cv2.imwrite("/content/drive/My Drive/Colab Notebooks/test_tif.tif",read)
```

6.2. EXPORT MODEL AND PERFORM UNIT TESTING

```
print(imageC1)
# test case 1:
#159A9_CRC-Prim-HE-07_022.tif_Row_901_Col_151.tif ; 1EAE_CRC-Prim-HE-
10_029.tif_Row_1_Col_451.tif   [tumor or debris]
```

```
#4B46_CRC-Prim-HE-07.tif_Row_301_Col_601.tif [tumor debris adipose]
#test_image=plt.imread("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/01_TUMOR/"+imageC1)
# test case 2:
test_image=plt.imread("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/04_LYMPHO/"+imageC1)
#test case 3:
#test_image=plt.imread("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/03_COMPLEX/17D73_CRC-Prim-
HE-01_034.tif_Row_451_Col_301.tif")
#print(os.listdir("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/02_STROMA"))
#test_image=plt.imread("/content/drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000/02_STROMA/11385_CRC-Prim-
HE-06_003.tif_Row_601_Col_151.tif")
plt.imshow(test_image)
file_name=[]
predictions=[]
from PIL import Image as PImage
import cv2
#from fastai.vision import *
#-----check for lympho---
lympholist=os.listdir("/content/drive/My Drive/Colab Notebooks/test_images/")
print(len(lympholist))
for i in range(0,len(lympholist)):
 if(lympholist[i].endswith(".jpeg")):
test_image=plt.imread("/content/drive/My Drive/Colab
Notebooks/test_images/"+lympholist[i])
#------check end for lympho-----
```

```
  frame = cv2.cvtColor(test_image,cv2.COLOR_BGR2RGB)
  pil_im = PImage.fromarray(frame)
  x = pil2tensor(pil_im ,np.float32)
  preds_num = learner.predict(Image(x))[2].numpy()
   #print(preds_num)
   if True:# preds_num[4]!=0 and
preds_num[4]==max([preds_num[0],preds_num[1],preds_num[2],preds_num[3],preds_n
um[4],preds_num[5],preds_num[6],preds_num[7]]) :
  #print(l,"\n",class_names)
  file_name.append(lympholist[i])
  predictions.append(preds_num)
  #print(lympholist[i])
  #print(preds_num)
  #break
  #from sklearn.externals import joblib
  #print(os.listdir("./drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000"))
#classifer = joblib.load("./drive/My Drive/Colab
Notebooks/Kather_texture_2016_image_tiles_5000//drive/My Drive/Colab
Notebooks/pkl_colorectal_CNN_model.pkl")
for fn,pre in zip(file_name,predictions):
print(fn,"\t",pre)
count=0
pridict=[]
for i in range(len(preds_num)):
  if(preds_num[i]!=0):
        count+=1
        #print(class_names[i+1]," ---> ",preds_num[i])
```

```
        pridict.append(class_names[i+1])
        #print(class_names[i+1],"\t")

print("According to our dataset the scan matches with:\n"," and ".join(pridict),"type of colorectal cancer")
```

6.3. IMPLEMENT MODEL TO PREDICT OVER WEB

```
from flask import Flask,render_template,request,flash,url_for,redirect
from werkzeug.utils import secure_filename
import json
import random
import tablib
l=learner.load_learner("./models/level1.pth")
app=Flask(__name__)
app.secret_key = 'h432hi5ohi3h5i5hi3o2hi'
#create a route
@app.route('/')
def home():
        return render_template('index.html')
@app.route('/prediction',methods=['GET','POST'])
def result():
        if request.method == 'POST':
        #flash(" ".join(request.form.keys()))
        f=request.form['img_file'].split("/")
        #-----------------------------------#
        #result=jsonify(l.predict(f))       #
        #json.dump(result,"testfile.json")   #
```

```
#-----------------------------------#
with open("testfile.json") as jfile:
dicl=json.load(jfile)
ifile=f[len(f)-1]
if ifile in dicl.keys():
result=dicl[ifile]
furl="/test_images/"+f[len(f)-1]
stage="Can't identify"
if 'Mucosa' in result:
stage="S0"
if 'Lympho' in result:
        stage="S1"
elif 'Debris' in result:
        stage="S2"

if 'Stroma' in result or 'Complex' in result:
        stage="S3I"
 elif 'Lympho' in result:
stage="S3"
if 'Stroma' in result or 'Complex' in result:
        stage="S3I"
 #print(cell)
 #print(result1)
return
render_template('index.html',isindex=True,imagef=str(url_for("static",filename=furl)),result=result,stage=stage)
else:
return redirect(url_for('home'))
```

```
@app.route('/model')
def model():
	return render_template('model.html')
```

PIP lock file:


```
{
	"_meta": {
	"hash": {
	"sha256":
"d5e270fef618e43f481bda408d839a4499afea70db57b11d01d3c414e4b94b4f"
	},
	"pipfile-spec": 6,
	"requires": {
	"python_version": "3.7"
	},
	"sources": [
	{
		"name": "pypi",
		"url": "https://pypi.org/simple",
		"verify_ssl": true
	}
	]
	},
	"default": {
	"beautifulsoup4": {
	"hashes": [
	"sha256:4c98143716ef1cb40bf7f39a8e3eec8f8b009509e74904ba3a7b315431577
e35",
```

"sha256:84729e322ad1d5b4d25f805bfa05b902dd96450f43842c4e99067d5e1369eb25",

"sha256:fff47e031e34ec82bf17e00da8f592fe7de69aeea38be00523c04623c04fb666"
],
"index": "pypi",
"version": "==4.9.3"
},
"click": {
"hashes": [

"sha256:d2b5255c7c6349bc1bd1e59e08cd12acbbd63ce649f2588755783aa94dfb6b1a",

"sha256:dacca89f4bfadd5de3d7489b7c8a566eee0d3676333fbb50030263894c38c0dc"
],
"markers": "python_version >= '2.7' and python_version not in '3.0, 3.1, 3.2, 3.3, 3.4'",
"version": "==7.1.2"
},
"flask": {
"hashes": [

"sha256:4efa1ae2d7c9865af48986de8aeb8504bf32c7f3d6fdc9353d34b21f4b127060",

"sha256:8a4fdd8936eba2512e9c85df320a37e694c93945b33ef33c89946a340a238
557"
],
"index": "pypi",
"version": "==1.1.2"
},
"itsdangerous": {
"hashes": [

"sha256:321b033d07f2a4136d3ec762eac9f16a10ccd60f53c0c91af90217ace7ba1f
19",

"sha256:b12271b2047cb23eeb98c8b5622e2e5c5e9abd9784a153e9d8ef9cb4dd09d749"
],
"markers": "python_version >= '2.7' and python_version not in '3.0, 3.1, 3.2, 3.3'",
"version": "==1.1.0"
},
"jinja2": {
"hashes": [

"sha256:03e47ad063331dd6a3f04a43eddca8a966a26ba0c5b7207a9a9e4e08f1b29
419",

"sha256:a6d58433de0ae800347cab1fa3043cebbabe8baa9d29e668f1c768cb87a33
3c6"
],

"markers": "python_version >= '2.7' and python_version not in '3.0, 3.1, 3.2, 3.3, 3.4'",
"version": "==2.11.3"
},
"markupsafe": {
"hashes": [
"sha256:cd5df75523866410809ca100dc9681e301e3c27567cf498077e8551b6d20e42f",
"sha256:3b8a6499709d29c2e2399569d96719a1b21dcd94410a586a18526b143ec8470f",
"sha256:9add70b36c5666a2ed02b43b335fe19002ee5235efd4b8a89bfcf9005bebac0d",
"sha256:b1282f8c00509d99fef04d8ba936b156d419be841854fe901d8ae224c59f0be5",
"sha256:ade5e387d2ad0d7ebf59146cc00c8044acbd863725f887353a10df825fc8ae21",
"sha256:b2051432115498d3562c084a49bba65d97cf251f5a331c64a12ee7e04dacc51b",
"sha256:09c4b7f37d6c648cb13f9230d847adf22f8171b1ccc4d5682398e77f40309235",

"sha256:b1dba4527182c95a0db8b6060cc98ac49b9e2f5e64320e2b56e47cb2831978c7",

"sha256:596510de112c685489095da617b5bcbbac7dd6384aeebeda4df6025d0256a81b",

"sha256:e8313f01ba26fbbe36c7be1966a7b7424942f670f38e666995b88d012765b9be",

"sha256:ba59edeaa2fc6114428f1637ffff42da1e311e29382d81b339c1817d37ec93c6",

"sha256:84dee80c15f1b560d55bcfe6d47b27d070b4681c699c572af2e3c7cc90a3b8e0",

"sha256:d53bc011414228441014aa71dbec320c66468c1030aae3a6e29778a3382d96e5",

"sha256:acf08ac40292838b3cbbb06cfe9b2cb9ec78fce8baca31ddb87aaac2e2dc3bc2",

"sha256:195d7d2c4fbb0ee8139a6cf67194f3973a6b3042d742ebe0a9ed36d8b6f0c07f",

"sha256:09027a7803a62ca78792ad89403b1b7a73a01c8cb65909cd876f7fcebd79b161",

"sha256:6f1e273a344928347c1290119b493a1f0303c52f5a5eae5f16d74f48c15d4a85",

"sha256:13d3144e1e340870b25e7b10b98d779608c02016d5184cfb9927a9f10c689f42",

"sha256:46c99d2de99945ec5cb54f23c8cd5689f6d7177305ebff350a58ce5f8de1669e",

"sha256:7fed13866cf14bba33e7176717346713881f56d9d2bcebab207f7a036f41b850",

"sha256:7c1699dfe0cf8ff607dbdcc1e9b9af1755371f92a68f706051cc8c37d447c905",

"sha256:9bf40443012702a1d2070043cb6291650a0841ece432556f784f004937f0f32c",

"sha256:e249096428b3ae81b08327a63a485ad0878de3fb939049038579ac0ef61e17e7",

"sha256:feb7b34d6325451ef96bc0e36e1a6c0c1c64bc1fbec4b854f4529e51887b1621",

"sha256:24982cc2533820871eba85ba648cd53d8623687ff11cbb805be4ff7b4c971aff",

"sha256:22c178a091fc6630d0d045bdb5992d2dfe14e3259760e713c490da5323866c39",

"sha256:43a55c2930bbc139570ac2452adf3d70cdbb3cfe5912c71cdce1c2c6bbd9c5d1",

"sha256:cdb132fc825c38e1aeec2c8aa9338310d29d337bebbd7baa06889d09a60a1fa2",

"sha256:6788b695d50a51edb699cb55e35487e430fa21f1ed838122d722e0ff0ac5ba15",

"sha256:1027c282dad077d0bae18be6794e6b6b8c91d58ed8a8d89a89d59693b9131db5",

"sha256:717ba8fe3ae9cc0006d7c451f0bb265ee07739daf76355d06366154ee68d221e",

"sha256:98bae9582248d6cf62321dcb52aaf5d9adf0bad3b40582925ef7c7f0ed85fceb",

"sha256:98c7086708b163d425c67c7a91bad6e466bb99d797aa64f965e9d25c12111a5e",

"sha256:29872e92839765e546828bb7754a68c418d927cd064fd4708fab9fe9c8bb116b",

"sha256:535f6fc4d397c1563d08b88e485c3496cf5784e927af890fb3c3aac7f933ec66",

"sha256:b7d644ddb4dbd407d31ffb699f1d140bc35478da613b441c582aeb7c43838dd8",

"sha256:62fe6c95e3ec8a7fad637b7f3d372c15ec1caa01ab47926cfdf7a75b40e0eac1",

"sha256:00bc623926325b26bb9605ae9eae8a215691f33cae5df11ca5424f06f2d1f473",

"sha256:2beec1e0de6924ea551859edb9e7679da6e4870d32cb766240ce17e0a0ba2014",

"sha256:caabedc8323f1e93231b52fc32bdcde6db817623d33e100708d9a68e1f53b26b",

"sha256:79855e1c5b8da654cf486b830bd42c06e8780cea587384cf6545b7d9ac013a0b",

"sha256:8defac2f2ccd6805ebf65f5eeb132adcf2ab57aa11fdf4c0dd5169a004710e7d",

"sha256:d9be0ba6c527163cbed5e0857c451fcd092ce83947944d6c14bc95441203f032",

"sha256:bf5aa3cbcfdf57fa2ee9cd1822c862ef23037f5c832ad09cfea57fa846dec193",

"sha256:c8716a48d94b06bb3b2524c2b77e055fb313aeb4ea620c8dd03a105574ba704f",

"sha256:88e5fcfb52ee7b911e8bb6d6aa2fd21fbecc674eadd44118a9cc3863f938e735",

"sha256:6dd73240d2af64df90aa7c4e7481e23825ea70af4b4922f8ede5b9e35f78a3b1",

"sha256:6fffc775d90dcc9aed1b89219549b329a9250d918fd0b8fa8d93d154918422e1",

"sha256:a6a744282b7718a2a62d2ed9d993cad6f5f585605ad352c11de459f4108df0a1",

"sha256:d73a845f227b0bfe8a7455ee623525ee656a9e2e749e4742706d80a6065d5e2c",

"sha256:500d4957e52ddc3351cabf489e79c91c17f6e0899158447047588650b5e69183",

"sha256:b00c1de48212e4cc9603895652c5c410df699856a2853135b3967591e4beebc2"
],
"markers": "python_version >= '2.7' and python_version not in '3.0, 3.1, 3.2, 3.3'",
"version": "==1.1.1"
},
"pillow": {
"hashes": [

"sha256:165c88bc9d8dba670110c689e3cc5c71dbe4bfb984ffa7cbebf1fac9554071d6",

"sha256:1d208e670abfeb41b6143537a681299ef86e92d2a3dac299d3cd6830d5c7bded",

"sha256:22d070ca2e60c99929ef274cfced04294d2368193e935c5d6febfd8b601bf865",

"sha256:2353834b2c49b95e1313fb34edf18fca4d57446675d05298bb694bca4b194174",

"sha256:39725acf2d2e9c17356e6835dccebe7a697db55f25a09207e38b835d5e1bc032",

"sha256:3de6b2ee4f78c6b3d89d184ade5d8fa68af0848f9b6b6da2b9ab7943ec46971a",

"sha256:47c0d93ee9c8b181f353dbead6530b26980fe4f5485aa18be8f1fd3c3cbc685e",

"sha256:5e2fe3bb2363b862671eba632537cd3a823847db4d98be95690b7e382f3d6378",

"sha256:604815c55fd92e735f9738f65dabf4edc3e79f88541c221d292faec1904a4b17",

"sha256:6c5275bd82711cd3dcd0af8ce0bb99113ae8911fc2952805f1d012de7d600a4c",

"sha256:731ca5aabe9085160cf68b2dbef95fc1991015bc0a3a6ea46a371ab88f3d0913",

"sha256:7612520e5e1a371d77e1d1ca3a3ee6227eef00d0a9cddb4ef7ecb0b7396eddf7",

"sha256:7916cbc94f1c6b1301ac04510d0881b9e9feb20ae34094d3615a8a7c3db0dcc0",

"sha256:81c3fa9a75d9f1afafdb916d5995633f319db09bd773cb56b8e39f1e98d908
20",

"sha256:887668e792b7edbfb1d3c9d8b5d8c859269a0f0eba4dda562adb95500f60d
bba",

"sha256:93a473b53cc6e0b3ce6bf51b1b95b7b1e7e6084be3a07e40f79b42e83503f
bf2",

"sha256:96d4dc103d1a0fa6d47c6c55a47de5f5dafd5ef0114fa10c85a1fd8e02162
84b",

"sha256:a3d3e086474ef12ef13d42e5f9b7bbf09d39cf6bd4940f982263d6954b13f6
a9",

"sha256:b02a0b9f332086657852b1f7cb380f6a42403a6d9c42a4c34a561aa4530d5
234",

"sha256:b09e10ec453de97f9a23a5aa5e30b334195e8d2ddd1ce76cc32e52ba63c8b
31d",

"sha256:b6f00ad5ebe846cc91763b1d0c6d30a8042e02b2316e27b05de04fa6ec831
ec5",

"sha256:bba80df38cfc17f490ec651c73bb37cd896bc2400cfba27d078c2135223c12
06",

"sha256:c3d911614b008e8a576b8e5303e3db29224b455d3d66d1b2848ba6ca83f9
ece9",

"sha256:ca20739e30325428713823448557 9b28cb0d524401f83d5129b5ff9d606cb0a8",

"sha256:cb192176b477d49b0a327b2a5a4979552b7a58cd42037034316b8018ac3ebb59",

"sha256:cdbbe7dff4a677fb555a54f9bc0450f2a21a93c5ba2b44e09e54fcb72d2bd13d",

"sha256:cf6e33d92b1526190a1de904df21663c46a456758c0424e4f947ae9aa6088bf7",

"sha256:d355502dce85ade85a2511b40b4c61a128902f246504f7de29bbeec1ae27933a",

"sha256:d673c4990acd016229a5c1c4ee8a9e6d8f481b27ade5fc3d95938697fa443ce0",

"sha256:dc577f4cfdda354db3ae37a572428a90ffdbe4e51eda7849bf442fb803f09c9b",

"sha256:dd9eef866c70d2cbbea1ae58134eaffda0d4bfea403025f4db6859724b18ab3d",

"sha256:f50e7a98b0453f39000619d845be8b06e611e56ee6e8186f7f60c3b1e2f0feae"
],

"index": "pypi",
"version": "==8.1.0"
},
"soupsieve": {
"hashes": [

"sha256:407fa1e8eb3458d1b5614df51d9651a1180ea5fedf07feb46e45d7e25e6d6cdd",

"sha256:d3a5ea5b350423f47d07639f74475afedad48cf41c0ad7a82ca13a3928af34f6"
],
"markers": "python_version >= '3.0'",
"version": "==2.2"
}’
"tablib": {
"hashes": [

"sha256:41aa40981cddd7ec4d1fabeae7c38d271601b306386bd05b5c3bcae13e5aeb20",

"sha256:f83cac08454f225a34a305daa20e2110d5e6335135d505f93bc66583a5f9c10d"
],
"index": "pypi",
"version": "==3.0.0"
},
"werkzeug": {
"hashes": [

"sha256:2de2a5db0baeae7b2d2664949077c2ac63fbd16d98da0ff71837f7d1dea3fd43",

"sha256:6c80b1e5ad3665290ea39320b91e1be1e0d5f60652b964a3070216de83d2e47c"
],
"index": "pypi",
"version": "==1.0.1"
}
},
"develop": {}
}

CHAPTER 7

PERFORMANCE ANALYSIS

CHAPTER 7
PERFORMANCE ANALYSIS

7.1 RESULTS & DISCUSSIONS

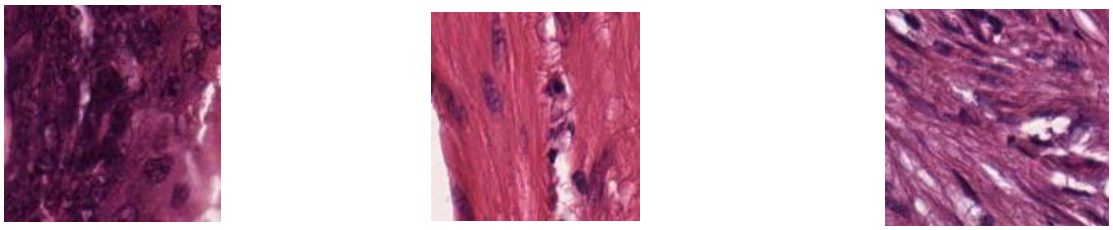
The training starts with dataset preparation and pre-processing and the data is separated using data splitting followed by modelling and then finally model deployment over web.

Dataset preparation and pre-processing:

Data is the foundation for any machine learningproject. The second stage of project implementation is complex and involves data collection, selection, preprocessing, and transformation which include:

Data collection:

The image is collected from the external source via endoscopic ultrasound which uses ultrasound imaging and endoscopy to determine abnormalities in the colon. We have collected around 5000 images i.e., under each tissue there are 645 images.



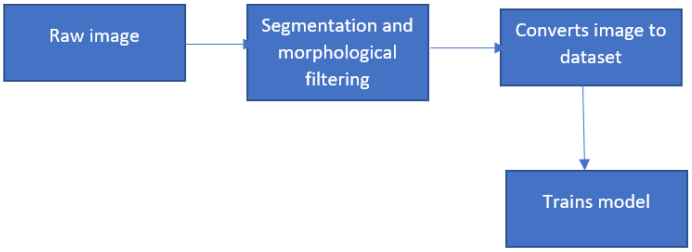


Fig. No. 7.1 Data Processing

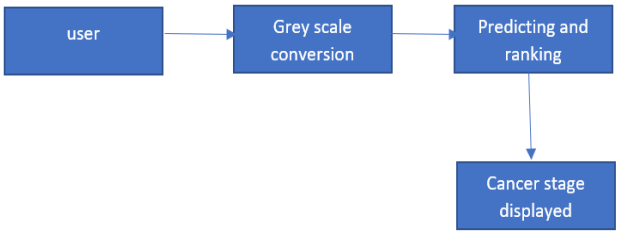


Fig. No. 7.2 Data Transformation

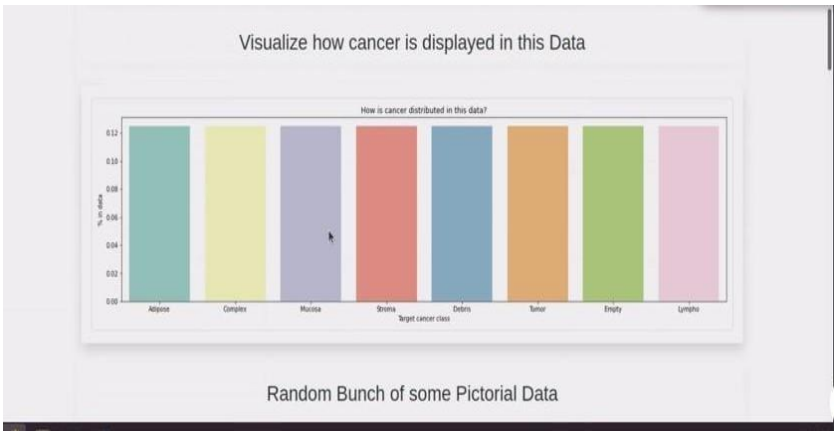


Fig. No. 7.3 Data Visualization

Data splitting:

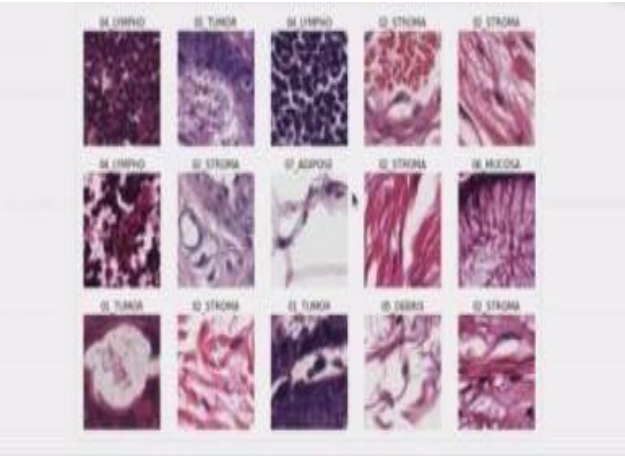


Fig. No. 7.4 Training set

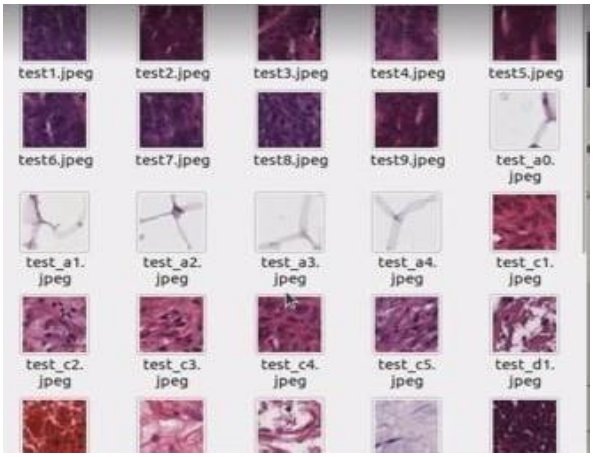


Fig. No. 7.5 Test set

Modelling:

During this stage we train numerous models to define which one of them provides the most accurate prediction.

Model training:

Here the human tissues collected from the lab/hospitals are to be drained and converted to datasets.

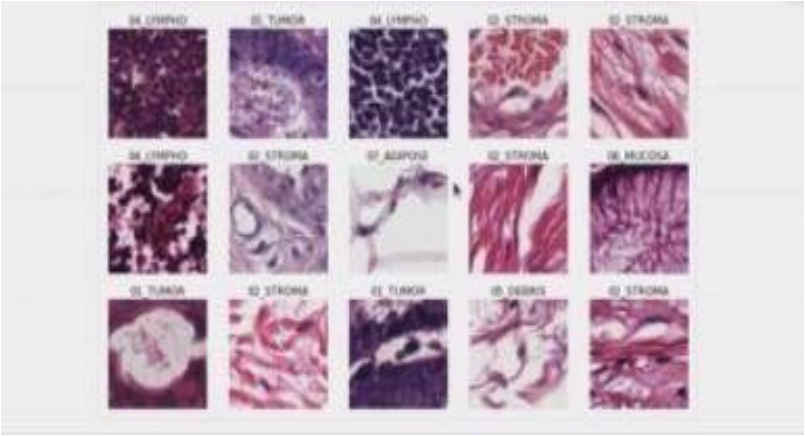


Fig. No. 7.6 Model Training

Model evaluation and testing:

This process is done by the user as the trained images are available and need to be selected and the stage of the cancer is predicted. This is done in the web page.

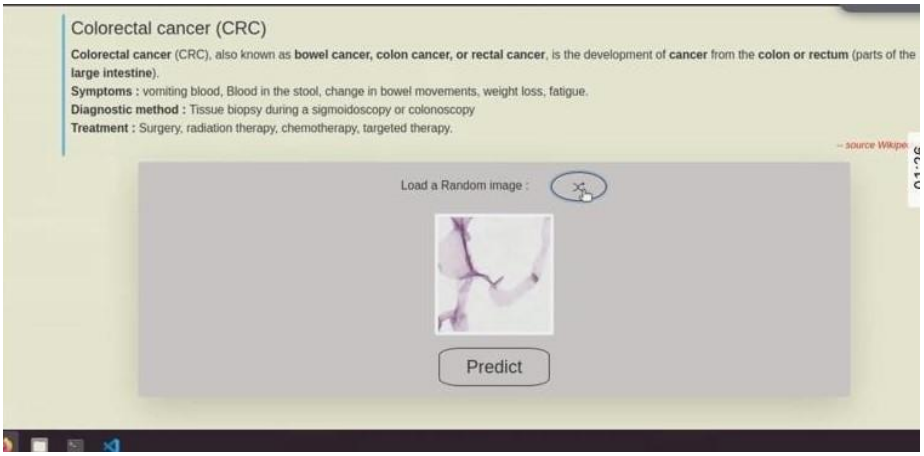


Fig. No. 7.7 Model evaluation and testing

Model deployment over web:

After selecting the image by the result as the cancer prediction and ranking is done over the web.

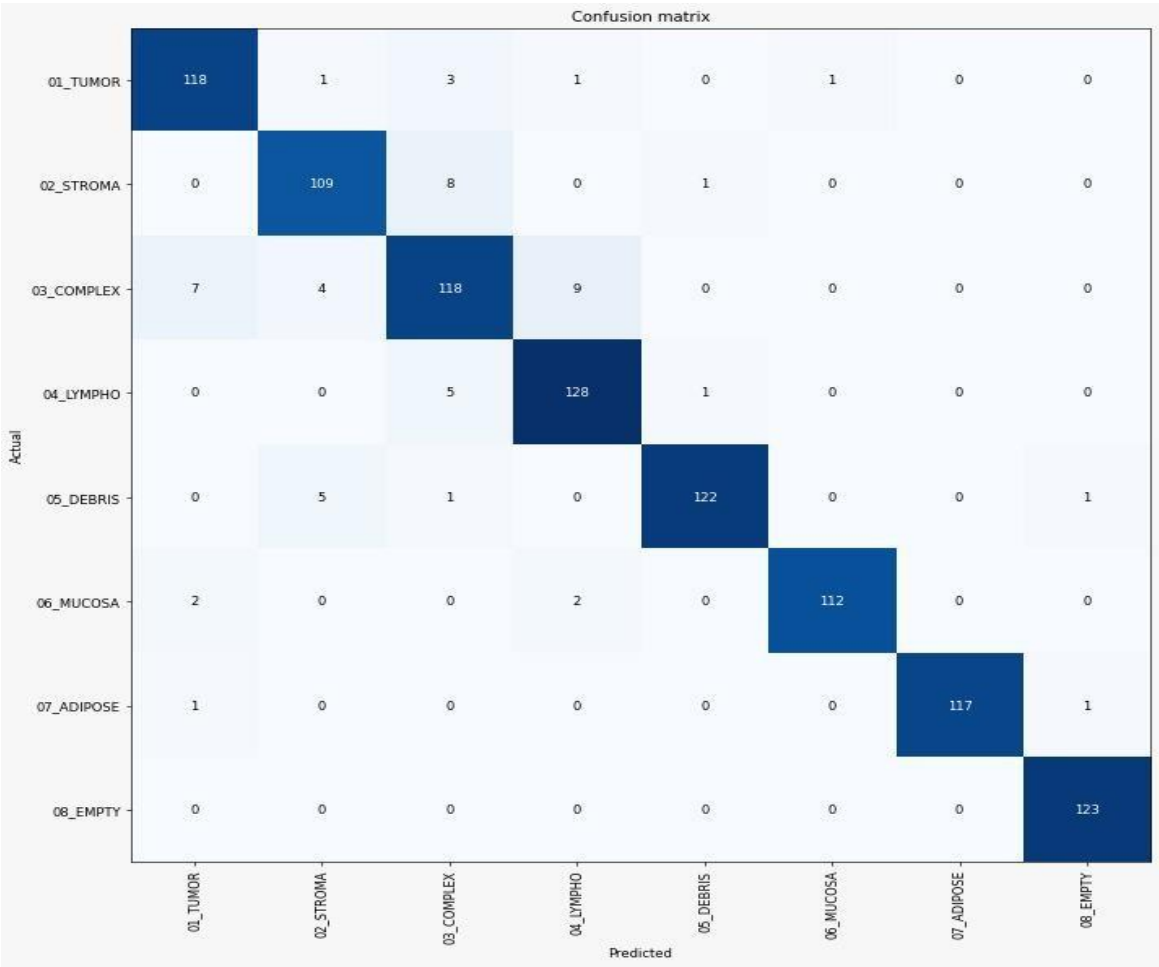


Fig. No. 7.8 Confusion Matrix

7.2 ACCURACY

Accuracy : It gives you the overall accuracy of the model, meaning the fraction of the total samples that were correctly classified by the classifier. To calculate accuracy, use the following formula: ***(TP+TN)/(TP+TN+FP+FN)***.

Misclassification Rate : It tells you what fraction of predictions were incorrect. It is also known as Classification Error. You can calculate it
using ***(FP+FN)/(TP+TN+FP+FN)*** or ***(1-Accuracy)***.

Precision: It tells you what fraction of predictions as a positive class were actually positive. To calculate precision, use the following formula: ***TP/(TP+FP)***.

Recall: It tells you what fraction of all positive samples were correctly predicted as positive by the classifier. It is also known as True Positive Rate (TPR), Sensitivity, Probability of Detection. To calculate Recall, use the following formula: ***TP/(TP+FN)***.

Specificity: It tells you what fraction of all negative samples are correctly predicted as negative by the classifier. It is also known as True Negative Rate (TNR). To calculate specificity, use the following formula: ***TN/(TN+FP)***.

F1-score: It combines precision and recall into a single measure. Mathematically it's the harmonic mean of precision and recall. It can be calculated as follows:

$$F_1\text{—score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Now, in a perfect world, we would want a model that has a precision of 1 and a recall of 1. That means a F1-score of 1, i.e. a 100% accuracy which is often not the case for a machine learning model. So what we should try, is to get a higher precision with a higher recall value.

epoch	train_loss	valid_loss	accuracy	time
0	0.839425	0.354545	0.881119	12:13
1	0.521362	0.425943	0.867133	12:15
2	0.478811	0.337351	0.899101	12:04
3	0.377444	0.358566	0.884116	12:06
4	0.328109	0.252855	0.915085	12:16
5	0.266330	0.210688	0.937063	12:05
6	0.208004	0.174468	0.946054	12:14
7	0.175398	0.172691	0.946054	12:14

Table 7.1 Accuracy with Epoch

CHAPTER 8

CONCLUSION AND FUTURE WORK

CHAPTER 8

CONCLUSION AND FUTURE WORK

Early detection of cancer is very important in the medical field. In this work, we present an image-based feature extraction, segmentation and training approaches for classification and screening of cancer tissues. The previous work focused on k-mean clustering which is less efficient and accuracy was 73%. In our work we have increased the accuracy to 94% by training the model with EPOCH. LSTM is used for fast processing and stores the best result for comparison in the future. Analysis is done by training a Neural network using the processed set of images in predicting the future output from input given to model. Future works Will Be Directed Towards Analysis of additional data sets acquired under controlled imaging conditions. Since the datasets under analysis in this work represent a huge variety of imaging condition variabilities, the observations from the experimental analysis are more generalizable yet limited in classification capabilities.

APPENDICES

A.1 SAMPLE SCREENS

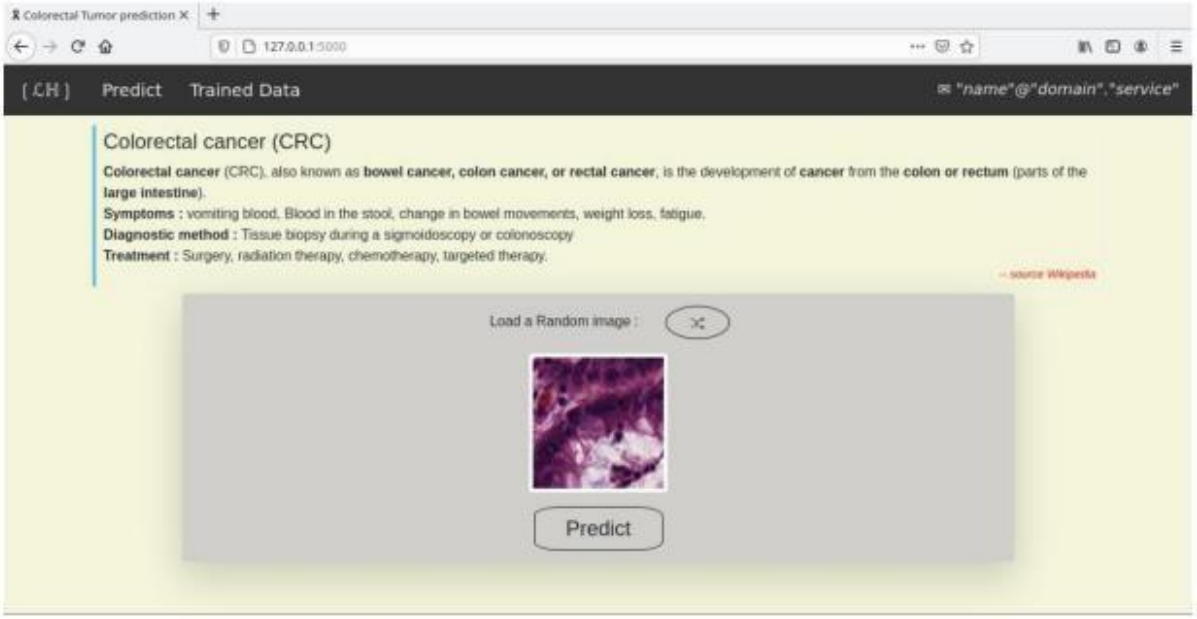


Fig 8.1 Uploading the image

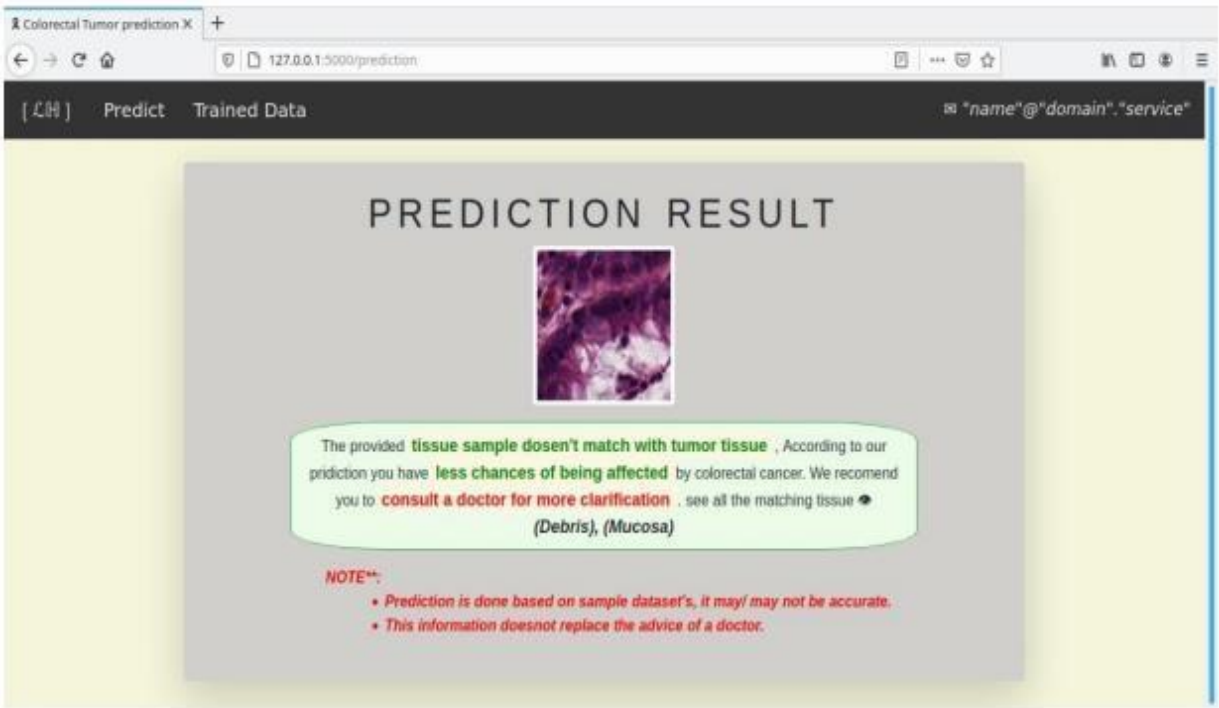


Fig 8.2 Result with less chance of colorectal cancer

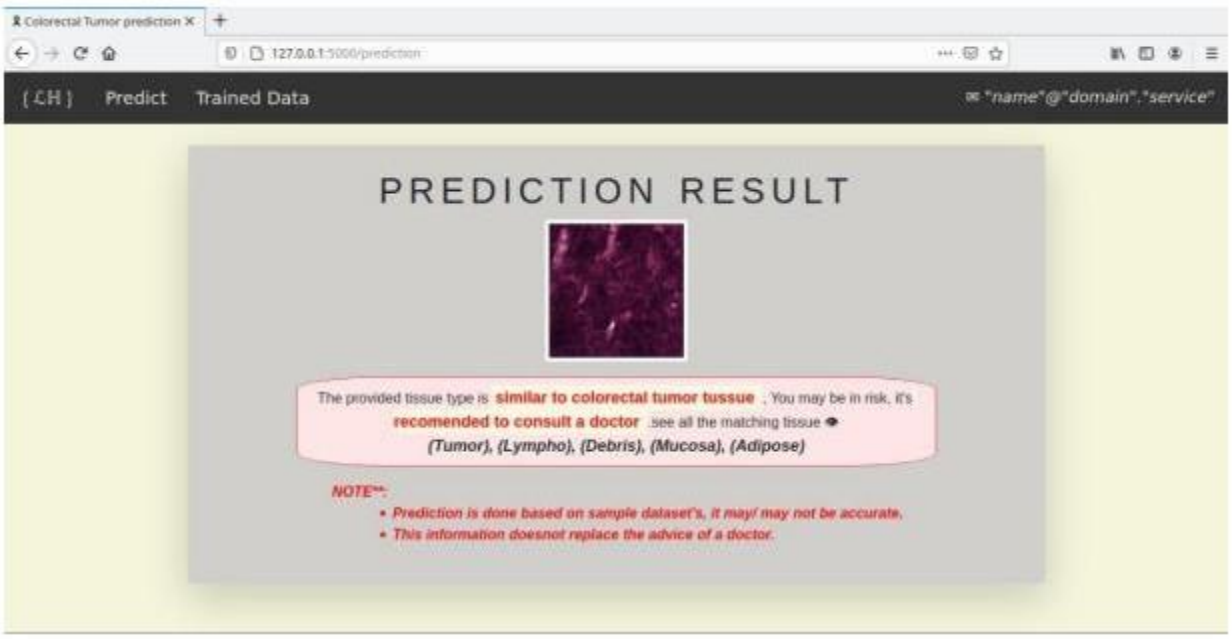


Fig 8.3 Result more chance of colorectal cancer

REFERENCE

[1]. Bunil Kumar Balabantaray,Kangkana Bora, Kunio Kasugai and Pallabi Sharma(2020)"Two Stage Classification With CNN For Colorectal Cancer Detection"Oncologie (SCI)

[2]. M. Jayakandan, T. Manivannan(February 2018) "Colorectal Cancer Detection in MRI Images Using Image Processing Techniques"International Journal of Engineering Science &Research Technology

[3].Jiri Prinosil,Malay Kishore Dutta, Namita Sengar, Neeraj Mishra, Radim Burget(2016)"Grading Of Colorectal Cancer Using Histology Images" 39th International Conference on Telecommunications and Signal Processing,29 June 2016

[4].KorsukSirinukunwattana et al.,(2016)"Locality Sensitive Deep Learning for Detection and Classification ofNuclei in Routine Colon Cancer Histology Images",IEEE Transactions on Medical Imaging(Volume:35, Issue: 5)

[5] Shi, G., Wang, Y., Zhang, C., Zhao, Z., Sun, X., Zhang, S., … Liu, J. (2018). Identification of genes involve in the four stages of colorectal cancer: Gene expression profiling. Molecular and Cellular Probes 37, 39-47. https://doi.org/10.1016/j.mcp.2017.11.004

[6] Huo, T., Canepa, R., Sura, A., Modave, F., & Gong, Y. (2017). Colorectal cancer stagestranscriptome analysis. PLoS ONE, 12(11), e0188697.http://doi.org/10.1371/journal.pone.0188697

[7] Siegel, R. L., Miller, K. D. and Jemal, A. (2017), Cancer statistics, 2017. CA: A Cancer Journal for Clinicians, 67: 7–30. doi:10.3322/caac.21387

[8] Tuan, J., & Chen, Y.-X. (2016). Dietary and Lifestyle Factors Associated with Colorectal Cancer Risk and Interactions with Microbiota: Fiber, Red or Processed Meat and Alcoholic Drinks. Gastrointestinal Tumors, 3(1), 17–24. http://doi.org/10.1159/000442831

[9] Vulcan, A., Manjer, J., Ericson, U., & Ohlsson, B. (2017). Intake of different types of redmeat, poultry, and fish and incident colorectal cancer in women and men: results from the Malmö Diet and Cancer Study. Food & Nutrition Research,61(1), 1341810. http://doi.org/10.1080/16546628.2017.1341810 4910. Helmus, D. S., Thompson, C. L., Zelenskiy, S., Tucker, T. C., & Li, L. (2013). Red meat-derived heterocyclic amines increaserisk of colon can- cer: a population-based case-control study. Nutrition and Cancer, 65(8), 1141–1150. http://doi.org/10.1080/01635581.2013.834945

[11] Cascella, M., Bimonte, S., Barbieri, A., Del Vecchio, V., Caliendo, D., Schiavone, V., Cuomo, A. (2018) Dissecting the mechanisms and molecules underlying the potential carcinogenicity of red and processed meat in colorectal cancer (CRC): an overview on the current state of knowledge. Infectious Agents and Cancer, 13(3), doi: 10.1186/s13027-018- 0174-9

[12] Riondino, S., Roselli, M., Palmirotta, R., Della-Morte, D., Ferroni, P., & Guadagni, F. (2014). Obesity and colorectal cancer: Role of adipokines in tumor initiation and progression.World Journal of Gastroenterology : WJG, 20(18), 5177–5190. http://doi.org/10.3748/wjg.v20.i18.5177

[13] Coleman, O. I., & Haller, D. (2017). Bacterial Signaling at the Intestinal Epithelial Interface in Inflammation and Cancer. Frontiers in Immunology, 8, 1927. http://doi.org/10.3389/fimmu.2017.01927

[14] Candela, M., Turroni, S., Biagi, E., Carbonero, F., Rampelli, S., Fiorentini, C., & Brigidi,

P. (2014). Inflammation and colorectal cancer, when microbiota-host mutualism breaks. World Journal of Gastroenterology : WJG, 20(4), 908–

922. http://doi.org/10.3748/wjg.v20.i4.908

[15] Rossi, M., Anwar, M.J., Usman, A., Keshavarzian, A., Bishehsari, F., (2018). ColorectalCancer and Alcohol Consumption-Populations to Molecules. Cancers : Cancers, 10(2) 38. doi:10.3390/cancers10020038

[16] Linhart, K.; Bartsch, H.; Seitz, H.K. The role of reactive oxygen species (ROS) and cytochrome P-450 2E1 in the generation of carcinogenic etheno-DNA adducts. Redox Biology. 2014, 3, 56–62.

[17] Roelands, J., Kuppen, P. J. K., Vermeulen, L., Maccalli, C., Decock, J., Wang, E., Hendrick, W. (2017). Immunogenomic Classification of Colorectal Cancer and Therapeutic Implications. International Journal of Molecular Sciences, 18(10), 2229. http://doi.org/10.3390/ijms18102229

[18] Shukla, P. K., Chaudhry, K. K., Mir, H., Gangwar, R., Yadav, N., Manda, B., Rao, R. (2016). Chronic ethanol feeding promotes azoxymethan