A Project Report on

Histopathologic Cancer Detection

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Computer Engineering

by

Ketan Anil Muddalkar(16102022)
Rohan Prashant Dhere(15102057)

Under the Guidance of

Prof. Pravin Prakash Adivarekar

Department of Computer Engineering
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Approval Sheet

This Project Report entitled ***"Histopathologic Cancer Detection"*** Submitted by ***"Ketan Anil Muddalkar"(16102022), "Rohan Prashant Dhere"(15102057),*** is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Prof. Pravin Prakash Adivarekar
Guide

Prof. Sachin H. Malave
Head of Department of Computer Engineering

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"Histopathologic Cancer Detection"*** submitted by ***"Ketan Anil Muddalkar" (16102022),"Rohan Prashant Dhere" (15102057)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Science***,to the University of Mumbai,is a bonafide work carried out during academic year 2019-2020.

Prof. Pravin Prakash Adivarekar
Guide

Prof. Sachin H. Malave
Head Department of Computer Engineering

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Ketan Anil Muddalkar 16102022)
(Rohan Prashant Dhere 15102057)

Date:

Contents

List of Figures

List of Tables

List of Abbreviations

CNN:	Convolutional Neural Network
API :	Application Program Interface
NN :	Neural Network
OS :	Operating System
GPU :	Graphics Processing Unit
ML :	Machine Learning
DL :	Deep Learning
AI :	Artificial Intelligence
ConvNet :	Convolutional Neural Network
AWS :	Amazon Web Services
EC2 :	Elastic Compute Cloud
RDS :	Relational Database Service
VPC :	Elastic Virtual Private Cloud
ReLU :	Rectified Linear Activation unit

Abstract

We are implementing an algorithm that will automatically identify metastatic cancer in small image patches of human tissues taken from larger digital pathology scans. This method will reduce the time required to detect cancer compared to traditional manual methods. These images will be pre-processed and augmented using Keras API before passing it to the convolutional neural network model to improve the training process.

In this project, we aim to classify metastatic cancer images with maximum accuracy using deep learning techniques like Convolutional Neural Networks(CNN). The dataset which we intend to use will contain 2 classes, class one will consist of images of cancer-free patients and class two will consist of images of a patient having cancer. The models will be built using Keras API and TensorFlow and will be trained on GPU. We expect to work with 2 or more models which will be ensembled to get better accuracy compared to a single model.

Since one model may miss out on some features whereas other models might detect those features and interim improve the results. For this, we will build CNNs from scratch and also be using transfer learning methods. After achieving the desired accuracy we intend to deploy the model on the web. For the web hosting Amazon web services are used. It offers reliable and cheap computational machines to end-users.

Chapter 1

Introduction

Digital pathology is a new, rapidly expanding field of medical imaging. In digital pathology, whole-slide scanners are used to digitize glass slides containing tissue specimens at high resolution (up to 160nm per pixel).

In this project, we will focus on the detection of micro- and macro-metastases in lymph nodes. Lymph node metastases occur in most cancer types (e.g. breast, prostate, colon). Lymph nodes are small glands that filter lymph, the fluid that circulates through the lymphatic system. Cancer is a group of diseases involving abnormal cell growth with the potential to invade or spread to other parts of the body.

Following are the symptoms observed:

- Lumps formation.
- Thickening or swelling of parts.
- Shortness of breath.
- Abnormal bleeding.
- Prolonged cough.
- Unexplained weight loss.
- Headache, seizures, or dizziness.
- Change in bowel habits or bladder function.
- White patches inside the mouth or white spots on the tongue.
- Extreme tiredness.

1.1 Objective

Histopathology refers to the examination of a biopsy or surgical specimen by a pathologist, after the specimen has been processed and histological sections have been placed onto glass slides. The goal of this project is to apply an open science approach to develop algorithms to detect cancer metastasis in lymph node images. This examination will be done by our deep learning model without human involvement. For any given input image of tissues, the model will try to classify it into two already predefined classes. The classes two being positive cancer detected or negative.

We aim to achieve the following objectives by the end of the project:

- Creating multiple Convolutional Neural Networks.
- Understanding the role of Data Pre-processing.
- Understanding how multiple convolutional neural networks interact with each other.
- Understanding how ensembling models perform better than a single model.
- Create a smooth and user-friendly system for cancer detection.
- Generating a medical report based on the result predicted by the model and the patient's medical history.

Chapter 2

Literature Review

In this project we aim to incorporate all the best methods in each step of creating a deep learning model. Research in this field has been in constant state of motion as cancer is a significantly serious disease in our community. Histopathology is the diagnosis and study of diseases of the tissues, and involves examining tissues and/or cells under a microscope. Histopathologists are responsible for making tissue diagnoses and helping clinicians manage a patients care.

If the histopathologists are not well-trained, this may lead to an incorrect diagnosis. Also, there is a lack of specialists, which keeps the tissue sample on hold for up to two months, for example, this occurs often in Norway [1]. With the onset of pattern recognition and machine learning, many handcrafted (engineered) features-based studies are proposed for classifying breast cancer histology images.

With the recent advances in image processing and machine learning, the classification and diagnosis of breast cancer using histology images attract much interest, and developing pattern recognition-based diagnostic systems became a necessity to help experts optimize diagnostic quality.

There have been several studies utilising transfer learning, especially with Convolutional Neural Networks to detect and classify cancerous regions in histopathology images. Convolutional Neural Network (CNN) is a well-known deep learning architecture inspired by the natural visual perception mechanism of the living creatures. State-of-the-art CNN models for image classification are judged by their performance on the ImageNet challenge.

CNN models for image classification are judged by their performance on the ImageNet challenge. First CNN model introduced was AlexNet in 2012, with modifications to the architecture that other CNN models have improved their performance[2]. In 2014 GoogLeNet, which was based on AlexNet, by reducing the number of parameters involved. In 2014 VGGNet was introduced, which achieved great performance because of the depth of the neural network. In ResNet was introduced, which utilizes batch normalization. The performance of some of these models on ImageNet is shown in Table.

Model	ImageNet Error %
AlexNet	*15.3%*
GoogLeNet	*6.7%*
VGG 16	*7.3%*
Inception V2	*5.6%*
Inception V3	*3.58%*
ResNet	*3.57%*
Inception- ResNet	*3.08%*
NASnet	*3.8%*

Table 2.1: Accuracy values

Convolutional Neural Network consists of three types of layers, a convolutional layer, pooling layer, and fully-connected layers. The convolutional layer learns feature representations of the input image. It is composed of several convolution kernels which are used to compute feature maps. Each neuron of a feature map is connected to neighbouring neurons in the previous layer. This is known as the neurons receptive field. The new feature map can be obtained by first convolving the input with a kernel and then applying an element-wise nonlinear activation function on the results. To generate each feature map, the kernel is shared by all spatial locations of the input. The complete feature maps are obtained by different types of kernels.[3]

Chapter 3

Problem Statement

Cancer is a leading cause of death worldwide, accounting for an estimated 9.6 million deaths in 2018. Cancer arises from the transformation of normal cells into tumor cells in a multistage process that generally progresses from a precancerous lesion to a malignant tumor if undetected. A pathologists report after reviewing a patients biological tissue samples is often the gold standard in the diagnosis of many diseases. The reviewing of pathology slides is a very complex task, requiring years of training to gain the expertise and experience to do well. There can be substantial variability in the diagnoses given by different pathologists for the same patient, which can lead to misdiagnosis. However, there can be many slides per patient, each of which is 10+ gigapixels when digitized at 40X magnification. This is a lot of data to cover, and often time is limited.

Cancer cells are hard to detect for human eyes even for a professional pathologist who has been in the medical scene for a while. We intend to focus on deep learning techniques that will be capable of identifying cancer cells up to micro-level and thus help in improving decision-making skills. Hence with the help of this deep learning method, we intend to speed up the process of cancer detection with high accuracy and hence contribute towards a reduction in the number of deaths caused due to cancer.

Chapter 4

Scope

The scope of this project is to deliver a model, with world-class accuracy, which can detect cancer from histopathology images and produce top notch results. The project will involve gathering data from hospitals or websites. Preprocessing data as per requirements to make it suitable for our model. The model will be compiled on Keras with Tensorflow as its backend.

We expect the model to output the class to which the given image will belong(i.e. Cancer detected or not). Furthermore, we are planning to deploy the model on a website which can act as an example for hospitals to create a similar portal for the benefits mentioned ahead.

Chapter 5

Technology Stack

5.1 Languages

- Python for data preprocessing, data visualization, building deep learning models, Performance measurement graphs.
- PHP/Python will be used as a backend of website and also to create medical report after diagnosingPHP/Python will be used as a backend of website and also to create medical report after diagnosing.
- HTML (Hypertext Markup Language) is the code that is used to structure a web page and its content. It is not a programming language, it is a markup language that defines the structure of the content.

  CSS stands for Cascading Style Sheets with an emphasis placed on Style, Documents stylepage layouts, colors, and fonts are all determined with CSS.
- SQL stands for Structured Query Language. SQL is a standard language for storing, manipulating, and retrieving data in databases. It is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e. data incorporating relations among entities and variables.

5.2 Libraries

- Tensorflow for machine learning applications such as neural networks. It is an end-to-end open source machine learning platform. TensorFlow is an interface for expressing machine learning algorithms and an implementation for executing such algorithms. A computation expressed using TensorFlow can be executed with little or no change on a wide variety of heterogeneous systems.

- Keras is a high-level neural networks API, running on top of Tensorflow.It focuses on enabling fast experimentation also it efficiently executes low-level tensor operations on CPU, GPU, or TPU. The simplest type of model is the Sequential model, a linear stack of layers and more advance is Functional model.

- Pandas is open source data analysis and manipulation tool, built on top of the Python programming language. It offers fast and efficient DataFrame object for data manipulation and tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format.

- Numpy is the fundamental package for scientific computing in Python. This library will make matrix operations faster compared to traditional matrix operations in python. At the core of the NumPy package, is the ndarray object. This encapsulates n-dimensional arrays of homogeneous data types, with many operations being performed in compiled code for performance. There are several important differences between NumPy arrays and the standard Python sequences.

- OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. It is used for preprocessing images.

- Bootstrap library will allow in creating responsive web pages. Additionally, Bootstrap requires jQuery to function. jQuery is an extremely popular and widely used JavaScript library, that both simplifies and adds cross browser compatibility to JavaScript. The ”grid” is the most essential aspects of the framework. It’s the basis on which the entire layout is created. Beyond that, Bootstrap’s core CSS will also add helpful styling to forms, tables, buttons, lists, and images, as well as fully functioning navigation bars.

5.3 Environments/ Tools

- Apache server will be used to host the entire website. An Apache Server is a web server application that delivers content such as HTML pages, multimedia and CSS Style sheets over the internet. Apache Web Server is optimized and can handle a large amount of traffic and data transfer on minimal hardware requirements.

- phpMyAdmin is a free and open-source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP. We can run MySQL queries, optimize, repair, and check tables, change collation, and execute other database management commands.

- Machine learning models will be trained on NVIDIA DGX supercomputer. Nvidia DGX is a deep learning system, architected for high throughput and high interconnect bandwidth to maximize neural network training performance. The core of the system is a complex of four Tesla V100 GPUs.

- AWS: Amazon Web Services offers reliable, scalable, and inexpensive cloud computing services. This platform is used for computation power as well as for database service. Amazon EC2 provides Virtual computing environments, known as instances Preconfigured templates known as Amazon Machine Images (AMIs) Various configurations of CPU, memory, storage, and networking capacity for instances, known as instance types.
  Amazon RDS provides database instance types - optimized for memory, performance, or I/O - and provides six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and SQL Server.

Chapter 6

Benefits for environment

Training several large machine learning models with approximately 100 million parameters will have a carbon footprint of roughly 600 kg also tuning them will require significant amounts of energy. To reduce this amount will be using an approach of transfer learning where pre-trained models can be used to train on a new dataset. This method is computationally inexpensive which can save a lot of energy and help the environment. Also, we are using cloud computing services to host our web application, Cloud providers are the leading drivers in emission savings, reducing technologys impact on the environment through resource virtualization, continuous innovation, and robust data centers.

Migrating to the cloud means fewer machines and less hardware, which translates into lower cooling and space requirements. The end result: lower energy costs. Undoubtedly, cloud adoption and improved efficiencies will become increasingly ubiquitous with enduring advancements in cloud technology and growing green awareness.

Chapter 7

Benefits for the Society

This project aims to provide world class accuracy in detecting cancer from histopathology images. We expect our project will have a positive impact on the society at the cost of low to almost zero resource consumption. This project can act as an asset to doctors who are working in hospitals with cancer patients. With the help of this project will speed up the process of cancer detection, and in-turn help the patient in getting the medical attention required at the earliest.

Chapter 8

Application

Our model will be trained on images of patients having cancer. The same model, with minor tweaks, can be modified to work with MRI and CT scan images as well. Thus expanding its horizon in the medical field. These models can later be deployed on web. This will help the patient and doctors and all the system administrators to maintain a complete record of all the patients in one single go. Each hospital can maintain a portal by the help of which patients can access their reports and can consult to doctors immediately. In rural areas, hospitals can set-up their facilities and can send qualified doctors with such high-tech solutions and help the patients there.

Chapter 9

Proposed System

9.1 Working

Implementing an algorithm to identify metastatic cancer in small image patches taken from larger digital pathology scans. These images were pre-processed and augmented before passing it to the neural network model in order to improve the training process. For the training we are using deep learning technique like Convolutional Neural Network(CNN). We are working with 3 best models which are ensembled to get better accuracy compared to a single model. The 3 models we have used are VGG16, EfficientNet B0 and EfficientNet B5. We have used transfer learning method to train these 3 models. After achieving desired accuracy we have deployed the model on the web.

Web application is used to get test images from patient. To implement this web application we are using Amazon Web Services which provides us with free of cost computational power for hosting the website.

The user will be first greeted with login portal on which he/she will have to login into their respective account. If the user is new to the system it'll be mandatory for him/her to create an account and then proceed forward.

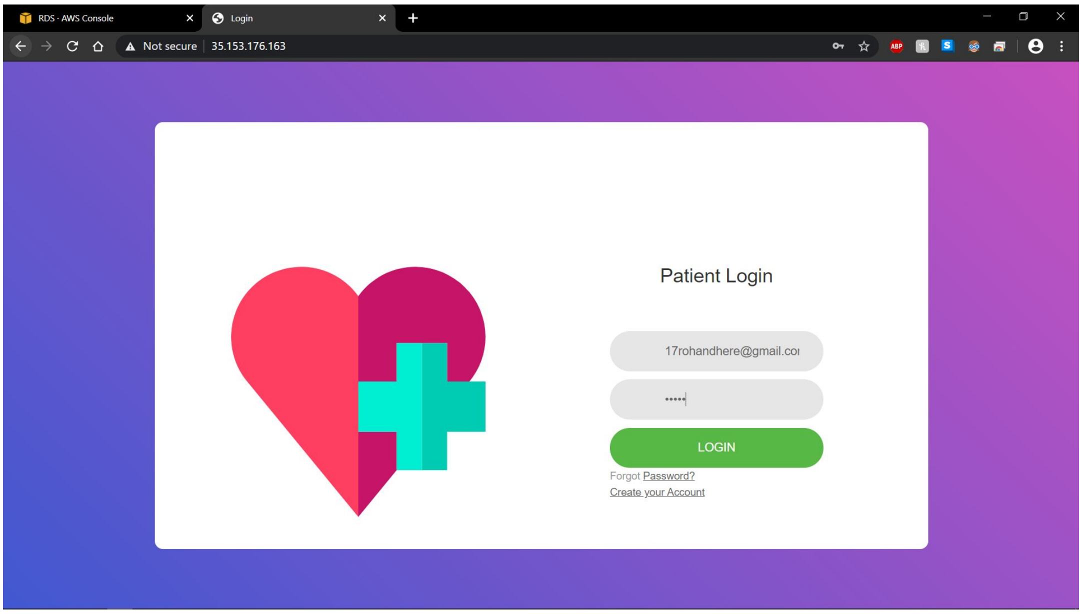


Figure 9.1: Login Page

If the user is new to the portal, he/she will have to create a new login account. For that all the details should be provided by the user, including medical history. After creating an account, the user will log into their account to upload the test image as shown below. This uploaded image will be fed to CNN model in the back-end. This model will is used to predict the outcome.

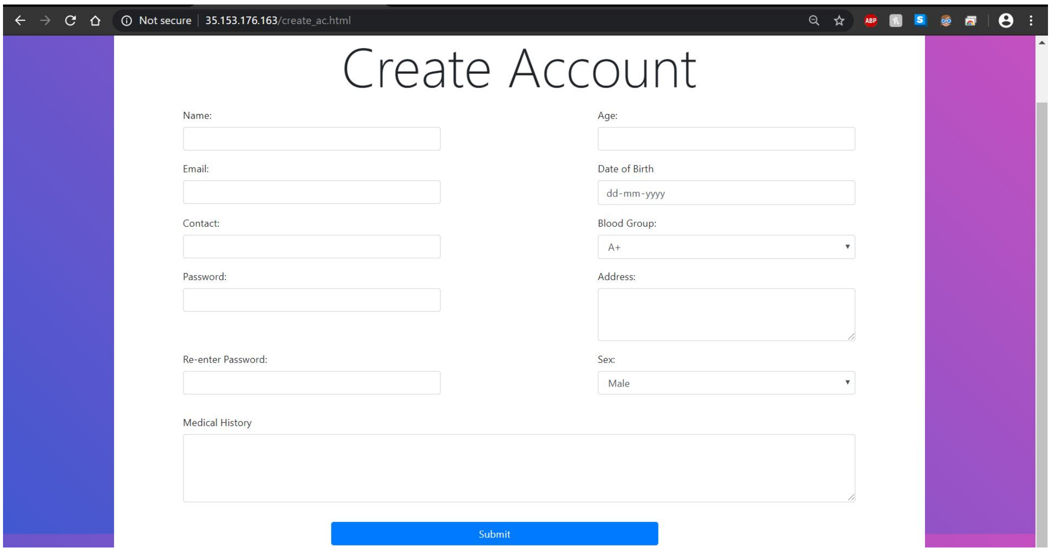


Figure 9.2: Create Account

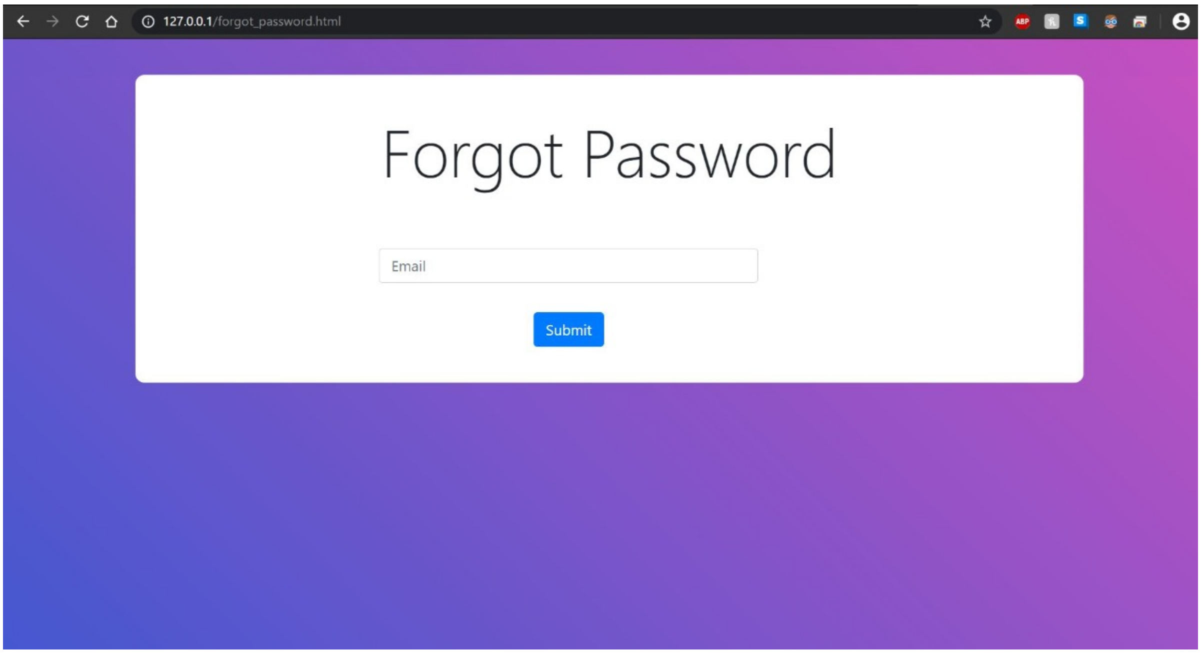


Figure 9.3: Forot Password Functionality

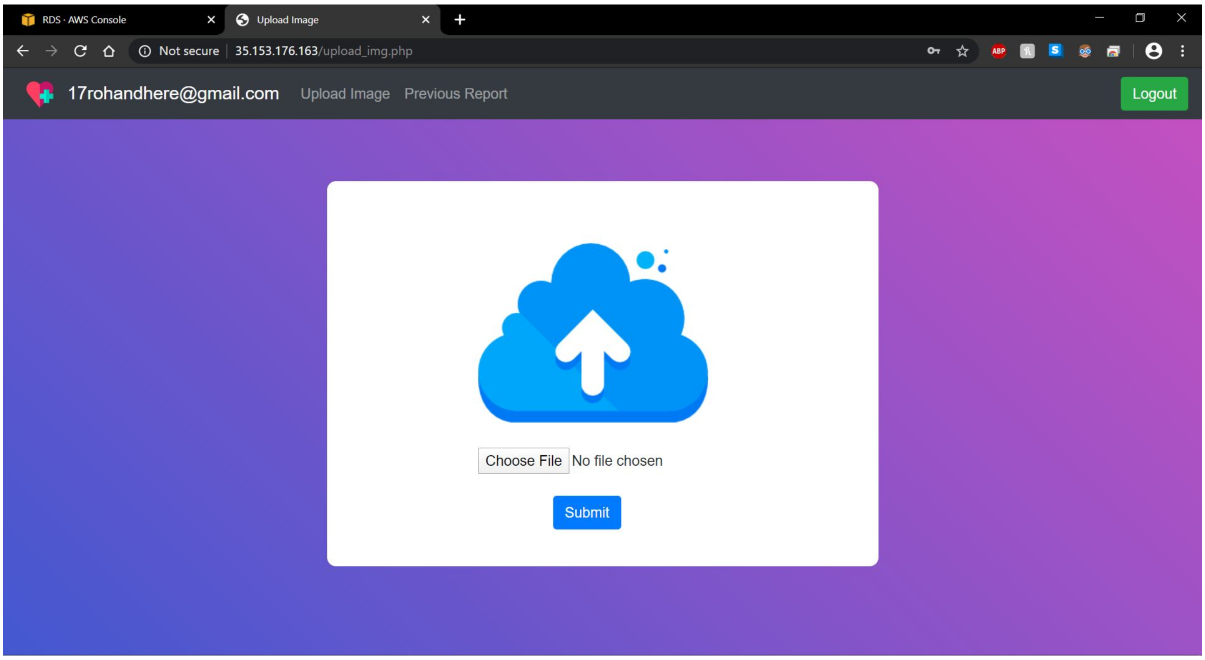


Figure 9.4: Upload Image

As soon as the result is calculated, the server generates a report of the outcome and is displayed to the user as well as stored in the records for future references. The report that is generated is shown below.

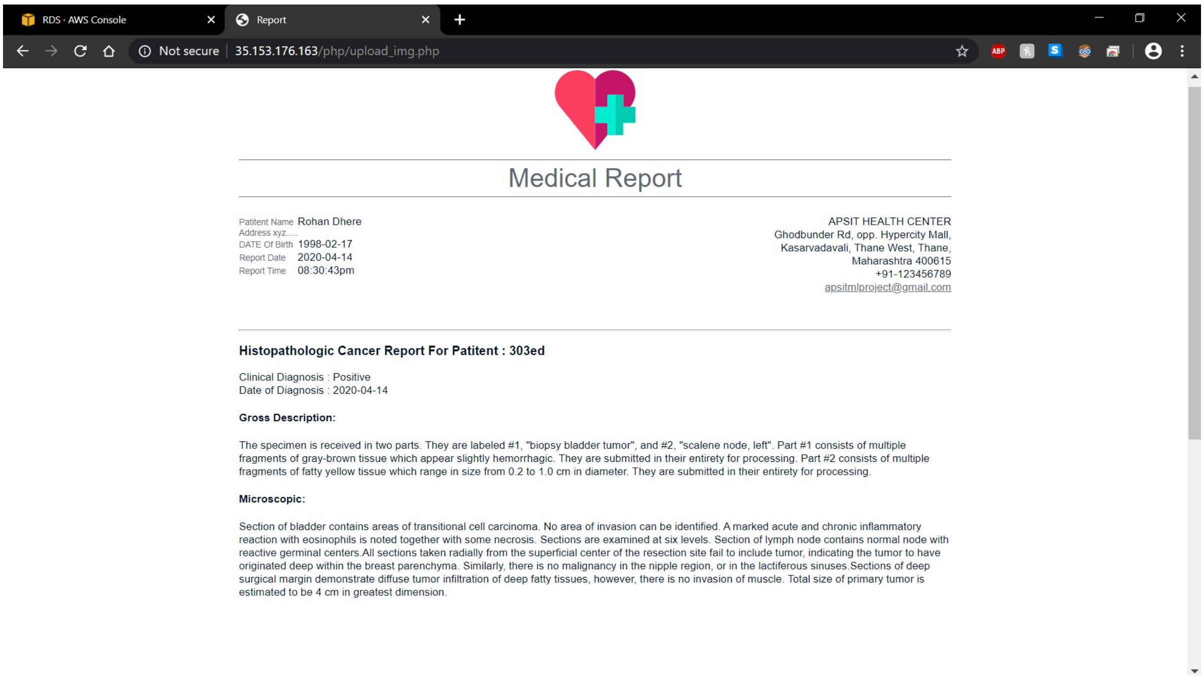


Figure 9.5: Generated Report

9.2 Activity Diagram

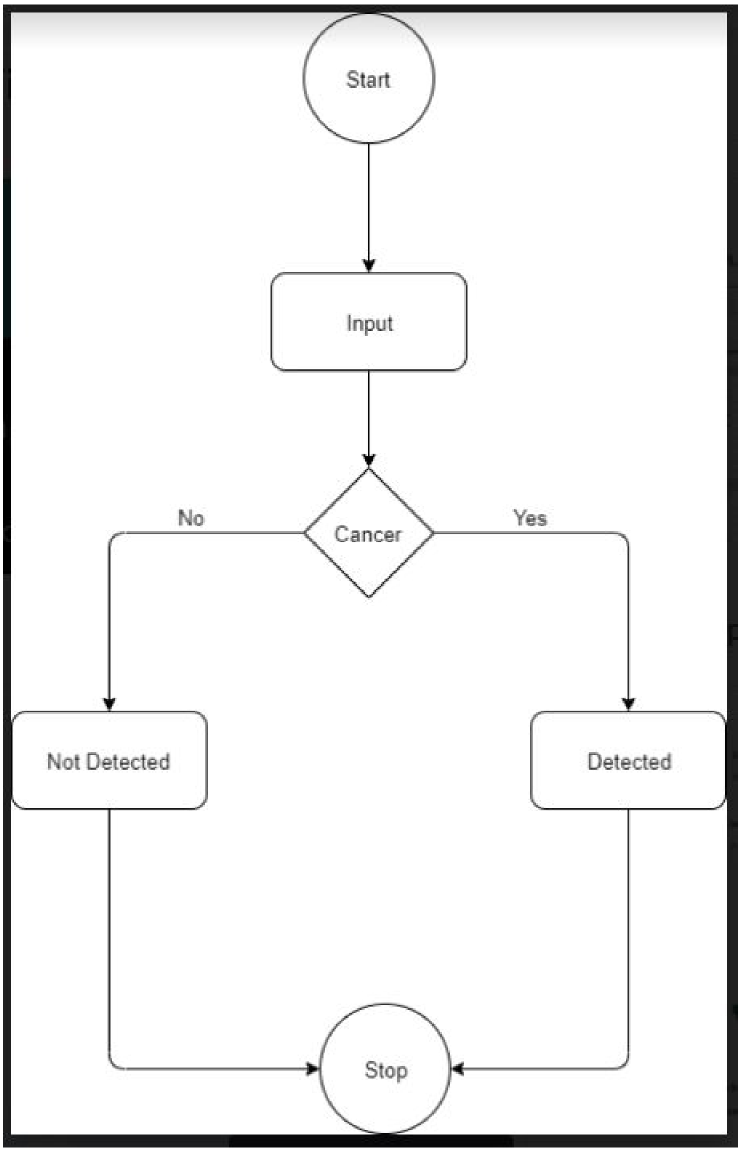


Figure 9.6: Activity Diagram

Chapter 10

Description of Use Case

The user will interact with the machine learning model via a portal. This portal will be an online repository which will allow user and the hospital to maintain proper records. The user will be requested to log-in into his/her account, after which he/she will be allowed to proceed further.

The user will upload necessary documents(in our case an image of CT/MRI scan of cancer affected area) which will be fed to the CNN model. The model will be allowed to make decision after properly studying the image. The generated result by the model will be represented in a proper report format and shown to the user in the portal itself. The copy of the report will also be mailed to the user for his/her personal safe keeping.

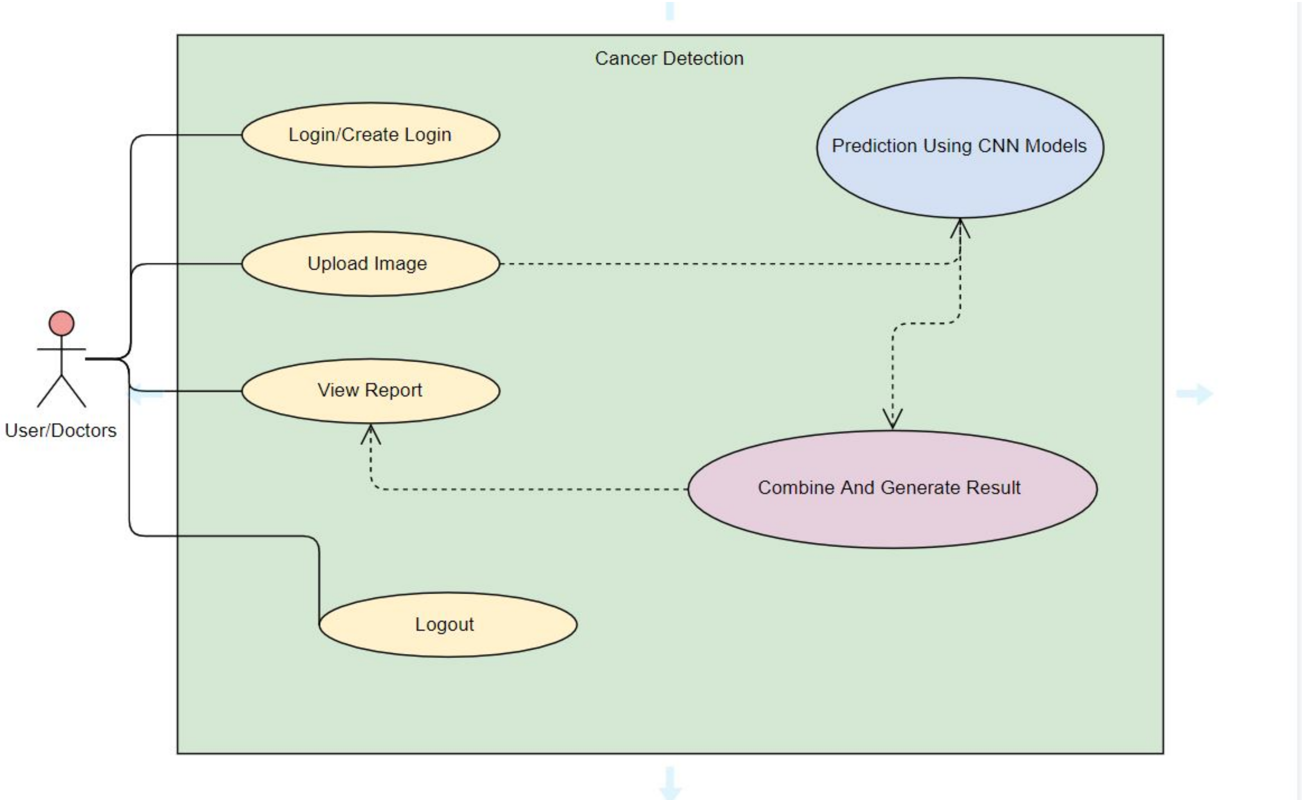


Figure 10.1: Use Case Diagram

Chapter 11

Modules

11.1 Front End

This module will focus on the front end aspect of the project. The designing of the web based portal will be briefly addressed here. The website will act as a portal for both hospital and patient. The motive of creating this website is to create a digital report for each patient which can be accessed anywhere.

The user, in our case a patient or doctor, will be required to create a user account and fill in all the details as mentioned. After successful creation of account the user will be logged in and his/her records will be generated there. From the account, the user will have the option to upload the reports for diagnosis. The uploaded document will be further passed on to CNN module which will be covered in further module.

Coming to the technologies being used, the front end will only include HTML, CSS and JavaScript. HTML and CSS will be used to create the website and design it. JavaScript will provide added functionalities to secure the website and make it fool-proof.

After the results have been computed by the CNN module, the server will generate a proper report which will contain all the necessary details. This report will be generated automatically by the server and a copy of it will be mailed to the user.

11.1.1 Hypertext Markup Language (HTML)

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript(JS).Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system. Berners-Lee specified HTML and wrote the browser and server software in late 1990.

The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991. It describes 18 elements comprising the initial, relatively simple design of HTML.

Advantages of HTML:

- HTML is free.
- Supported by all browsers.
- HTML can be integrated with other languages easily.
- HTML is lightweight.
- Compatible with a majority of development tools.
- HTML is the most search friendly programming language.

11.1.2 Cascading Style Sheet (CSS)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.

This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

CSS was first proposed by Hkon Wium Lie on October 10, 1994. At the time, Lie was working with Tim Berners-Lee at CERN. Several other style sheet languages for the web were proposed around the same time, and discussions on public mailing lists and inside World Wide Web Consortium resulted in the first W3C CSS Recommendation (CSS1) being released in 1996.

Advantages of CSS:

- Greater consistency in design.
- Provides more formatting options.
- Lightweight code.
- Ease of presenting different styles to different viewers.
- Reduces the file size and bandwidth usage.
- Faster load time.

11.1.3 Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden.

After a few months of development by a small group, many developers at Twitter began to contribute to the project as a part of Hack Week, a hackathon-style week for the Twitter development team. It was renamed from Twitter Blueprint to Bootstrap, and released as an open source project on August 19, 2011.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents.

Advantages of Bootstrap:

- Bootstrap is equipped with a responsive layout.
- Consistency.
- Simple Integration.
- Lightweight and customizable.
- A consistent framework that supports the majority of all browsers and CSS compatibility fixes.
- Fewer Cross browser bugs.

11.2 Dataset

11.2.1 Understanding the Data

This module will describe the dataset which is used to train the neural network model.The first step in creating a CNN model is to understand and work on the dataset. This is where data pre-processing comes into the picture.

The dataset is a slightly modified version of the PatchCamelyon (PCam) benchmark dataset (the original PCam dataset contains duplicate images due to its probabilistic sampling, however, the version presented on Kaggle does not contain duplicates). PCam is highly interesting for both its size, simplicity to get started on, and approach-ability. The dataset contains a total of 2.2 lakh images of cancer cells. All of the images are 92x92 pixels in size.

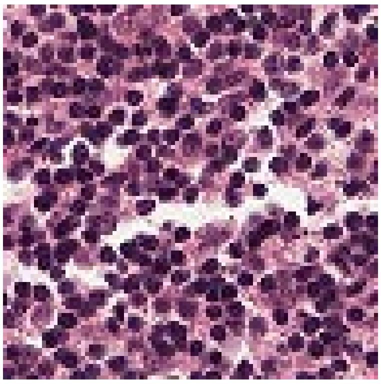


Figure 11.1: Healthy Cells

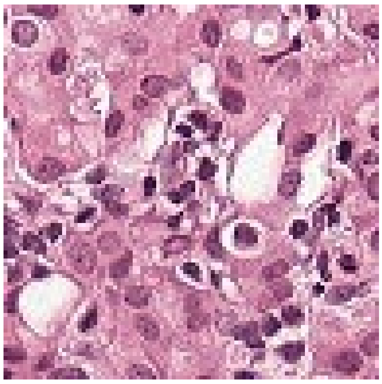


Figure 11.2: Cancer Cells

11.2.2 Data Pre-processing

This is where data pre-processing comes into the picture. One of the necessary pre-processing step is intensity normalization. The primary purpose of image normalization is to obtain the same range of values for each input image before feeding it to the CNN model which also helps to speed up the convergence of the model. Input images are normalized to the standard normal distribution by min-max normalization to the intensity range of [0, 1], which is computed as:

$$z = \frac{x \ - \min(x)}{\max(x) - \min(x)}$$

Figure 11.3: Normalization

Data augmentation is a strategy that enabled us to significantly increase the diversity of data available for training models, without actually collecting new data. Data augmentation techniques such as cropping, horizontal flipping, vertical flipping, rotation, shear are used to train large neural networks.

The table shown below represents the parameters we used during pre-processing:

Parameter	**Value**
Horizontal Flip	*True*
Vertical Flip	*True*
Rotation Range	*90*
Zoom Range	*0.2*
Width Shift Range	*0.1*
Height shift Range	*0.1*
Shear Range	*0.05*

Table 11.1: Data augmentation parameters

- Rotation range: Int. Degree range for random rotations.
- Width shift range: Float, 1-D array-like or int - float: fraction of total width
- Shear range: Float. Shear Intensity (Shear angle in counter-clockwise direction in degrees)
- Zoom range: Float or [lower, upper]. Range for random zoom.
- Horizontal flip: Boolean. Randomly flip inputs horizontally.
- Vertical flip: Boolean. Randomly flip inputs vertically.

Keras flow from directory method returns a DirectoryIterator yielding tuples of (x, y) where x is a numpy array containing a batch of images with shape (batch size, *target size, channels) and y is a numpy array of corresponding labels.

Below are some of the results that we obtained after performing pre-processing techniques on our dataset:

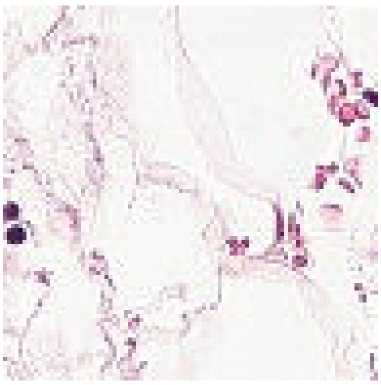


Figure 11.4: Horizontal flip

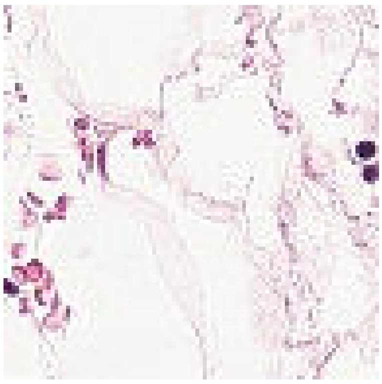


Figure 11.5: Vertical Flip

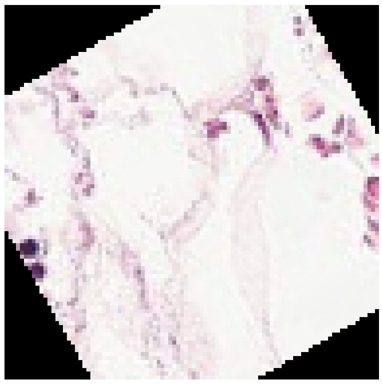


Figure 11.6: Random Rotation

11.3 Convolutional Neural Network

11.3.1 CNN Models

Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification. ConvNets have been successful in identifying faces, objects and traffic signs apart from powering vision in robots and self driving cars.

There are four main operations in the ConvNet

- Convolution
- Non Linearity (ACTIVATION)
- Pooling or Sub Sampling
- Classification (Fully Connected Layer)

These operations are the basic building blocks of every Convolutional Neural Network.

This module will describe the brain of the project. This model will be responsible for predicting the result from the image given by the user. The models will be trained on the images which were preprocessed. Among many models, the top three models were selected which are having the highest testing accuracy.These models are VGG16, EfficientNetB0, EfficientNetB5.

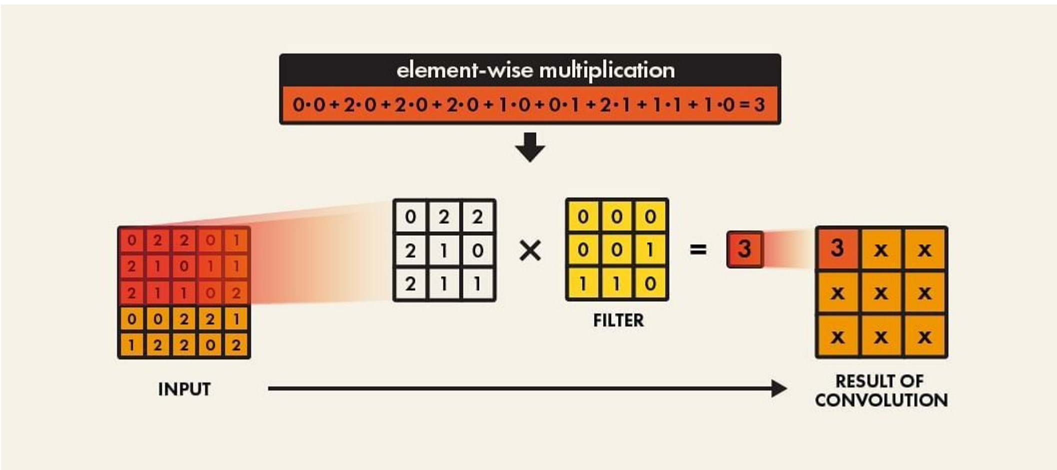


Figure 11.7: Convolution Step

VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford. The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. Another two models are from the EfficientNet family. EfficientNetB0 and EfficientNetB5.

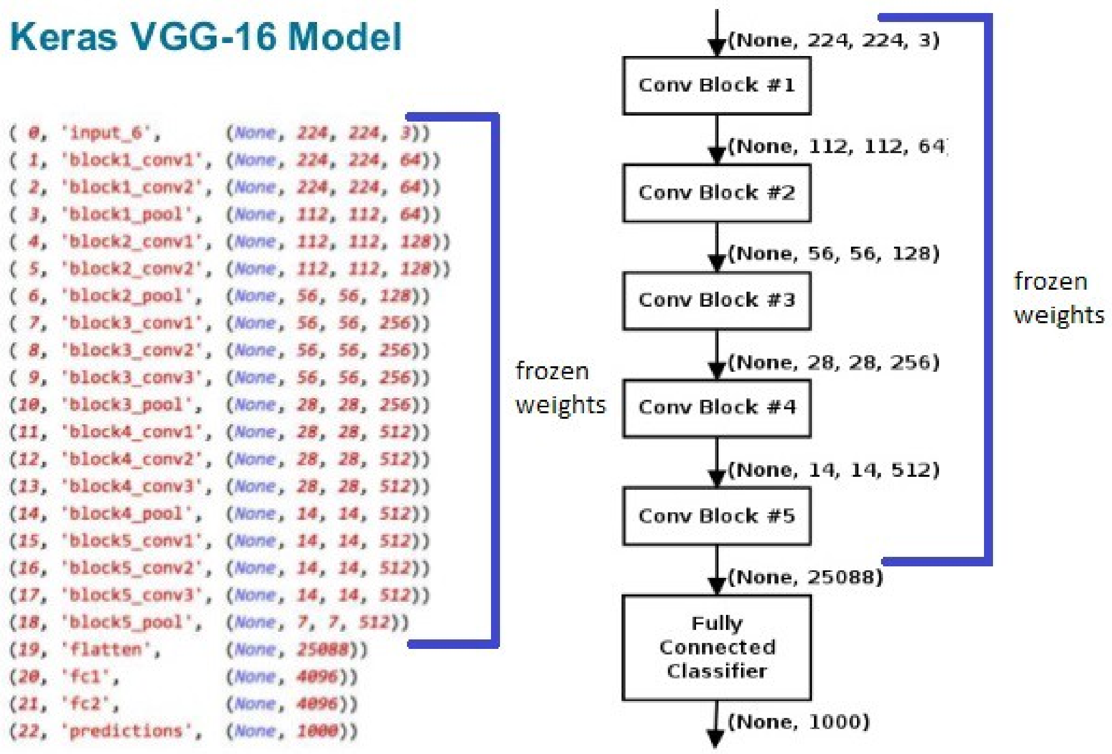


Figure 11.8: VGG16 Architecture

```
x = {OUTPUT OF CONVOLUTIONAL LAYERS / Flattened Features}

x = Dense(32)(x)                          // 32 fully connected neurons
x = BatchNormalization()(x)               // Regularization technique
x = Activation('relu')(x)                 // ReLU Activation Function
x = Dropout(0.25)(x)                      // Randomly Sets some 25% units to 0

x = Dense(16)(x)
x = BatchNormalization()(x)
x = Activation('relu')(x)
x = Dropout(0.25)(x)

x = Dense(8)(x)
x = BatchNormalization()(x)
x = Activation('relu')(x)
x = Dropout(0.25)(x)

x = Dense(4)(x)
x = BatchNormalization()(x)
x = Activation('relu')(x)
x = Dropout(0.25)(x)

predictions = Dense(1, activation='sigmoid')(x)      // Final Output Layer with one unit

model= Model(input = model.input, output = predictions)
```

Dense/Fully Connected Layers

Figure 11.9: Lower Levels of VGG16

The core idea about Efficient Nets is the use of compound scaling - using a weighted scale of three inter-connected hyper parameters of the model - Resolution of the input, Depth of the Network, and Width of the Network. When phi, the compound the coefficient is initially set to 1, we get the base configuration - in this case, EfficientNetB0.

These Three models are ensembled to get higher testing accuracy. After achieving the desired accuracy, the models are allowed to predict the outcome for the input provided by the user. Models will be kept in the back-end server and the input will be fed directly from the website as described.

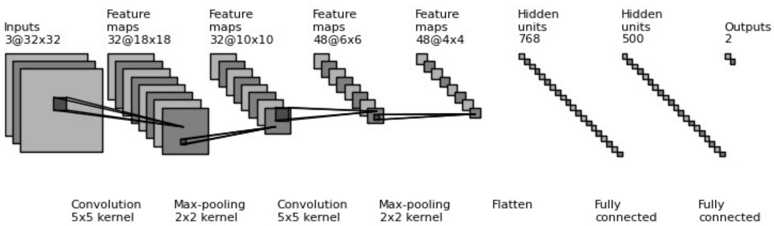


Figure 11.10: CNN Network Representation

11.3.2 Activation Functions

In a neural network, the activation function is responsible for transforming the summed weighted input from the node into the activation of the node or output for that input.

The rectified linear activation function is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero.

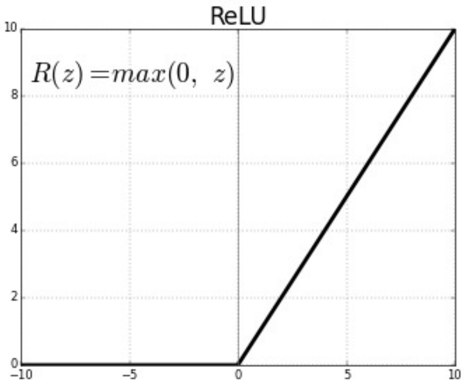


Figure 11.11: ReLU Graph

$$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$$

Figure 11.12: ReLU Formula

Sigmoid function is an activation function defined as a squashing function, which limits the output to a range between 0 and 1.

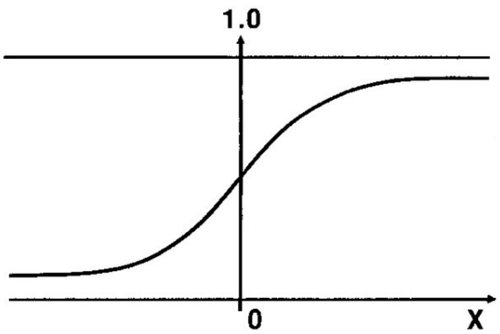


Figure 11.13: Sigmoid Graph

$$f(x) = \frac{1}{1 + e^{-(x)}}$$

Figure 11.14: Sigmoid Formula

11.3.3 Visualization

Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed. Python offers multiple great graphing libraries that come packed with lots of different features.

Convolutional neural networks have internal structures that are designed to operate upon two-dimensional image data, and as such preserve the spatial relationships for what was learned by the model. Specifically, the two-dimensional filters learned by the model can be inspected and visualized to discover the types of features that the model will detect, and the activation maps output by convolutional layers can be inspected to understand exactly what features were detected for a given input image.

Both filters and feature maps can be visualized. For example, we can design and understand small filters, such as line detectors. Perhaps visualizing the filters within a learned convolutional neural network can provide insight into how the model works. Given below are some examples taken from our own CNN models:

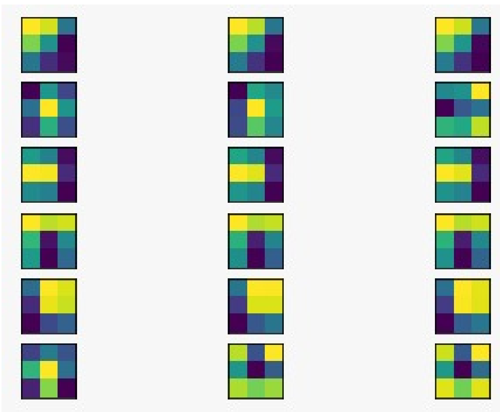


Figure 11.15: Filters Visualization

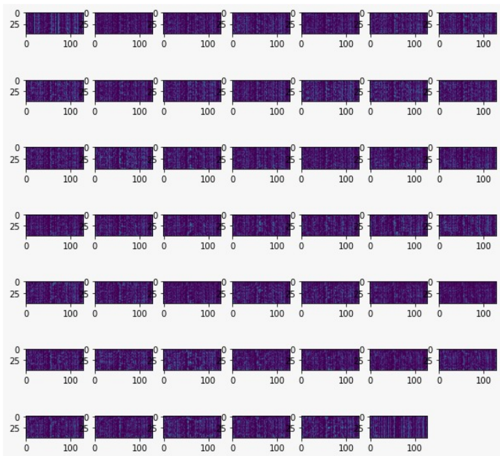


Figure 11.16: Filters Applied to Images

11.3.4 Experimental Setting

When all the images were fed to the CNN model without any preprocessing, 50% to 60% accuracy was achieved. The batch size was set from 32 to 521. In the case of DGX for the VGG16 model, even 1024 batch sizes were tested. A fully connected layer trained with the rectified linear unit (ReLU) activation function followed by a dropout layer with a probability from 0.2 to 0.5 were tested to prevent over-fitting. The dropout layer helps to further reduce overfitting by randomly eliminating their contribution to the training process. For Adam optimizer the learning rate was set to 0.1 to 0.0001.

All pre-trained Deep CNN models are fine-tuned separately. Also, the network weights were initialized from weights trained on ImageNet. The operating system was Ubuntu Desktop Linux OS with an Intel Xeon E5-2698 v4 2.2 GHz processor and GPUs were 4X Tesla V100. The training and testing process of the proposed architecture for this experiment is implemented in Python using Keras package with Tensorflow as the deep learning framework backend. Some Training was done on Googles colab.

11.4 Back End

The backend will be the backbone of this project. Itll contain the model, the database with all the user information and will be responsible for the generation of users medical report.

This will be a simple sql server containing some tables with users information(such as login id and password, email ids and more), the model and a simple block of code to generate a medical report. The backend will facilitate the movement of data within the server.

11.4.1 Computational Power

For the computational power we are using Amazon web services. It offers Elastic Compute Cloud (Amazon EC2) which is a web service that provides secure, resizable compute capacity in the cloud. T2 Mirco instance is a low-cost, general-purpose instance type that provides a baseline level of CPU performance with the ability to burst above the baseline when needed. It provides one CPU, One GB of RAM.

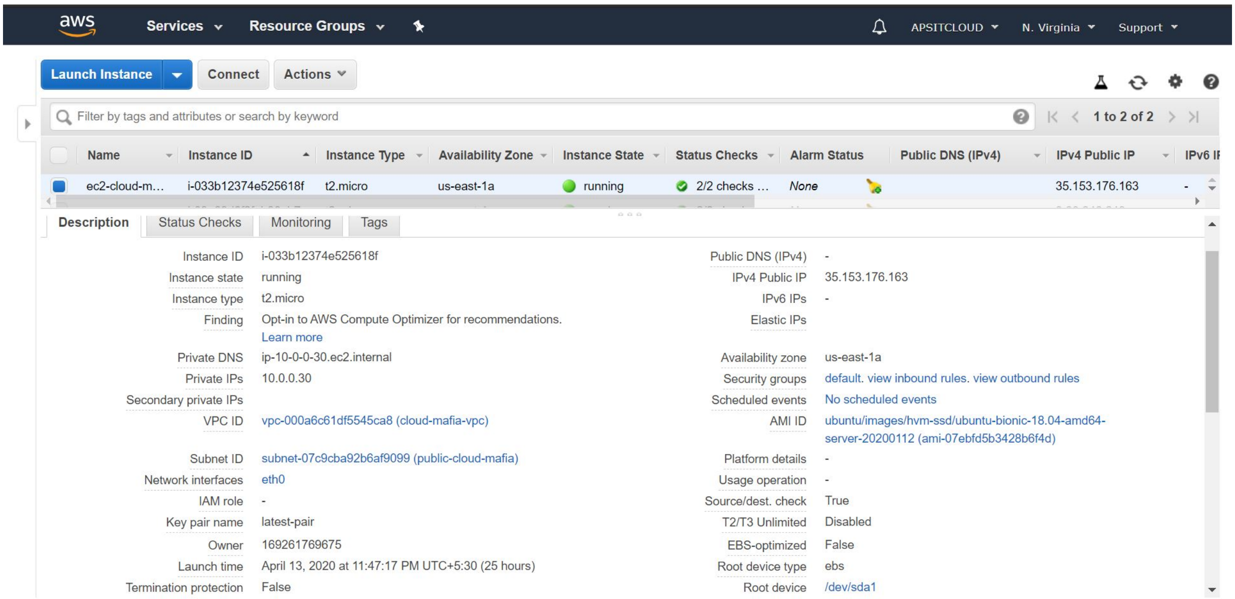


Figure 11.17: EC2 Instance

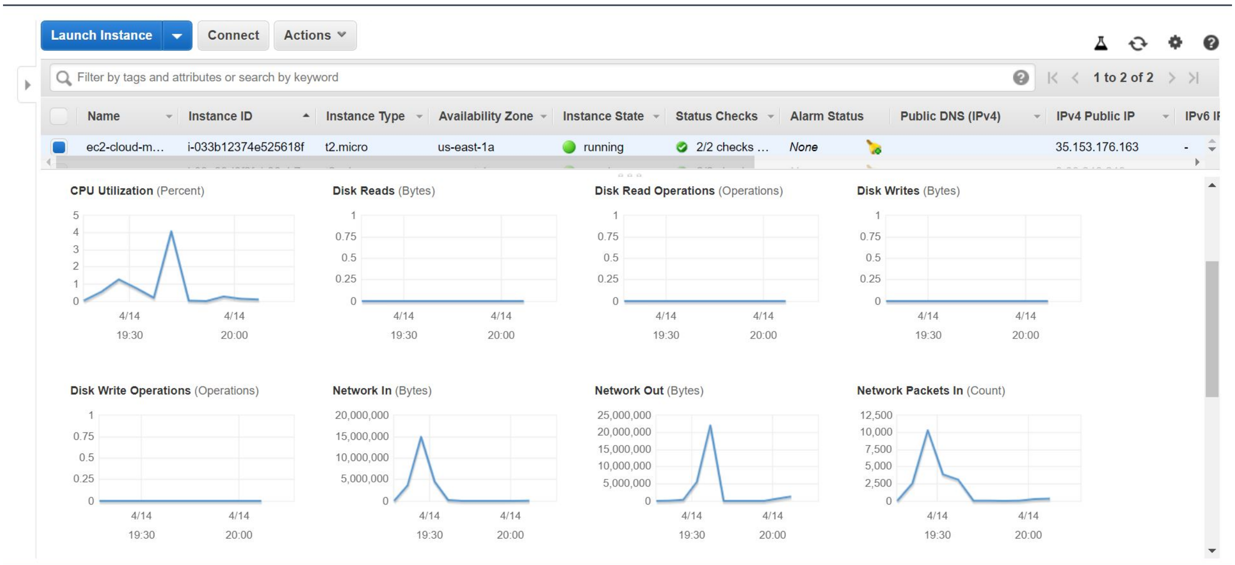


Figure 11.18: EC2 Usage Report

11.4.2 Virtual Private Cloud (VPC)

Amazon Virtual Private Cloud (Amazon VPC) helps to launch AWS resources into a virtual network. This virtual network closely resembles a traditional network with the benefits of using the scalable infrastructure of AWS. It provides on-demand configurable pool of shared computing resources allocated within a public cloud environment, providing a certain level of isolation between the different organizations.

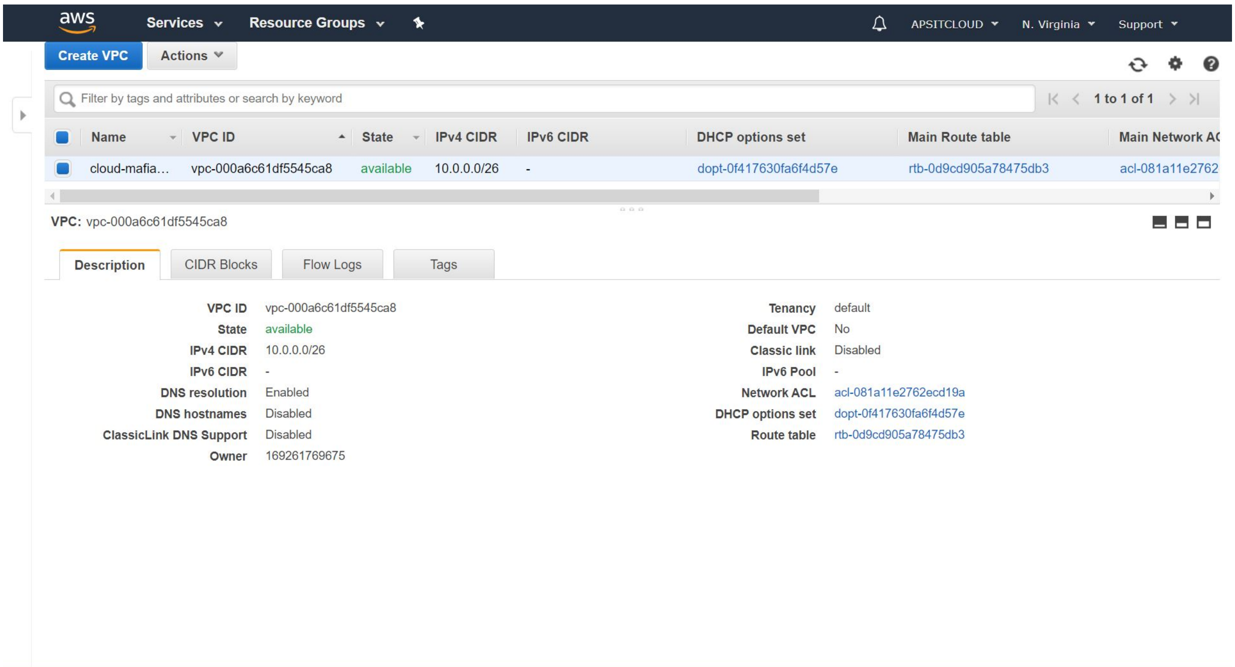


Figure 11.19: VPC Configuration

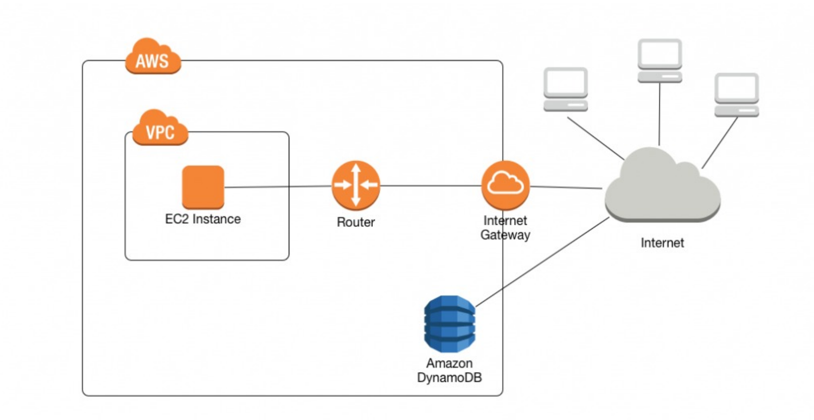


Figure 11.20: VPC Working

11.4.3 Operating System

Before launching an instance, we must select an AMI(Amazon Machine Image) to use. It provides the information required to launch an instance, which is a virtual server in the cloud. We are using Ubuntu 18.04 LTS Bionic which is available free of cost.

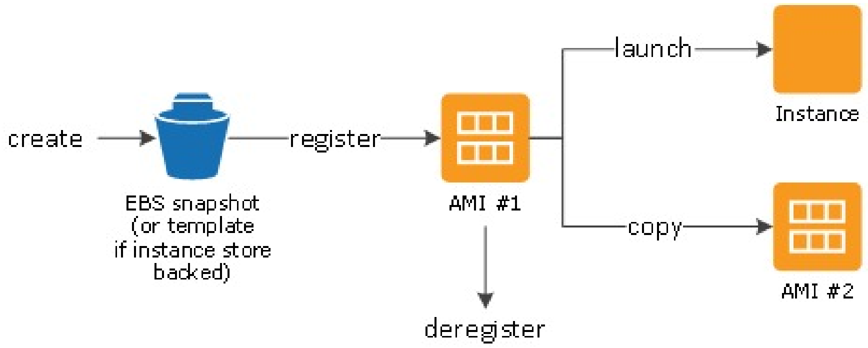


Figure 11.21: Process of Launching Instance

11.4.4 Web Server

Apache HTTP Server is a free and open-source web server that delivers web content through the internet. It processes requests and serves web assets and content via HTTP.

11.4.5 Languages

PHP is used as a server-side scripting language that is used to create dynamic web pages that can interact with databases. Python language is used for training of neural network and to analyze the images provided by the user by passing these images to neural networks.

11.4.6 Database

For the database, we are using Amazon's Relational Database Service (Amazon RDS). It is use to set up, operate, and scale a relational database in the cloud. MySQL is the world's most popular open-source relational database and Amazon RDS makes it easy to set up, operate, and scale MySQL deployments in the cloud.

The following images show the configuration of RDS on Amazon Cloud:

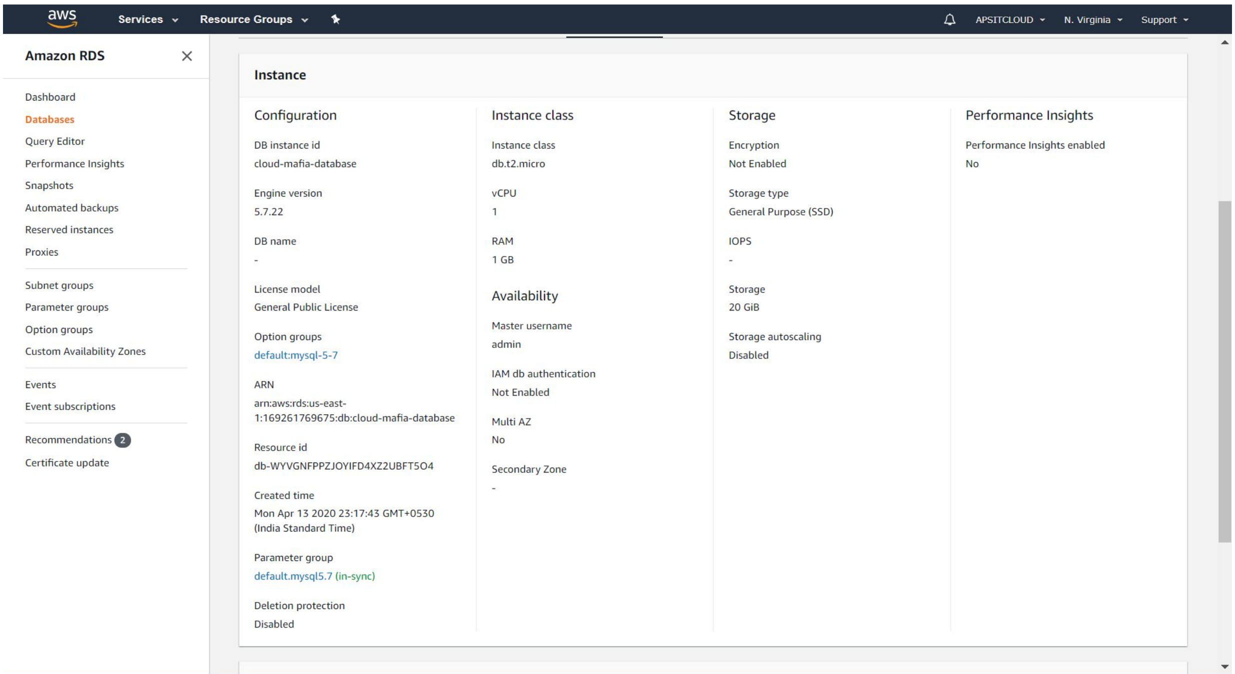
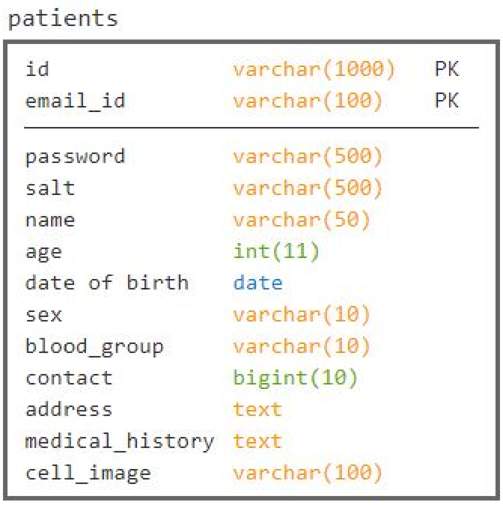


Figure 11.22: RDS Configuration

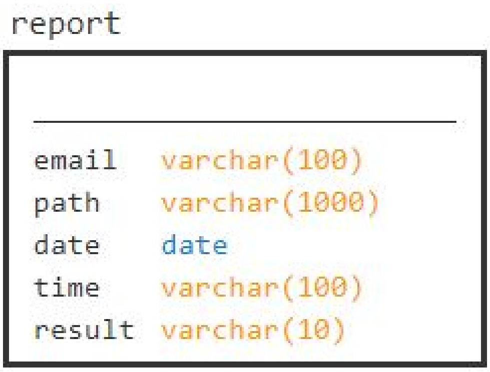
In this system we have used 2 tables to store all the information. Patients is the first table, and as the name suggests it stores all the details about the patients.



patients

id	varchar(1000)	PK
email_id	varchar(100)	PK
password	varchar(500)	
salt	varchar(500)	
name	varchar(50)	
age	int(11)	
date of birth	date	
sex	varchar(10)	
blood_group	varchar(10)	
contact	bigint(10)	
address	text	
medical_history	text	
cell_image	varchar(100)	

Figure 11.23: Healthy Cells



report

email	varchar(100)
path	varchar(1000)
date	date
time	varchar(100)
result	varchar(10)

Figure 11.24: Cancer Cells

Chapter 12

Result

The performance of the proposed classification model evaluated based on recall, precision, F1-score, and accuracy. Given the number of true positives (TP), false positives (FP), true negatives (TN) an false negatives (FN), the measures are mathematically expressed as follows:

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

Figure 12.1: Accuracy Criteria

We have used 3 CNN architecture, VGG16, Efficient net B0 and Efficient B5. Each one of the model has been tweaked to achieve maximum efficiency. As table show below depicts the accuracy of each of the model individually. All these 3 models were ensembeled and their cumulative accuracy was 96%.

Model	Train Acc	Test Acc
VGG16	95%	94%
EfficientNet B0	96%	94%
EfficientNet B5	95%	94%

Figure 12.2: Accuracy Table

Below are 2 samples taken from the dataset and the results are as follows:-

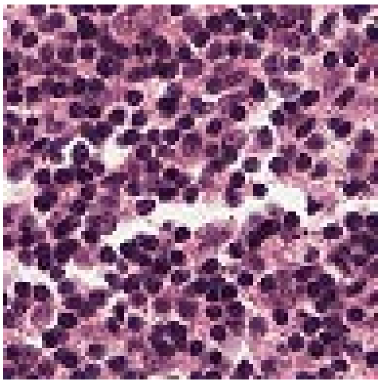


Figure 12.3: Class 0

Expected: Healthy Cells.

Predicted: Healthy Cells.

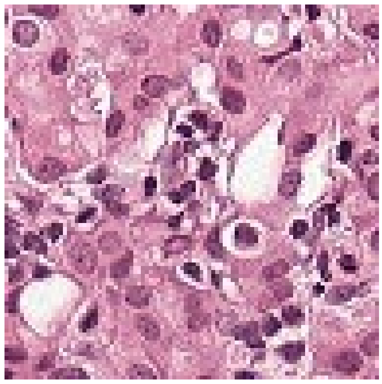


Figure 12.4: Class 1

Expected: Cancer Cells.

Predicted: Cancer Cells.

Chapter 13

Conclusions

Individual accuracies of convolutional neural network models were low compared to the accuracy of the ensembled network's accuracy. Because one CNN model may fail to detect some features but other CNN models can detect those features because of its different architecture and parameters. This means we should use more than one model to get our final results more accurate. As we have concluded that neural networks tend to miss out on some features, oncologists sometimes also miss some features and both the situations can lead to false results. It is not recommended that one should completely rely on this system and thus our proposed system should act as an assistant rather than a decision-maker.

Compared to a pathologist our system can generate results faster also automated detection of lymph node metastasis has great potential to help the pathologist and reduce their workload. Within the past few years, the field has been moving towards grand goals with strong potential diagnostic impact: (fully) automated analysis of whole-slide images to detect or grade cancer, to predict prognosis, or identify metastases.

Chapter 14

Future Scope

In most situations, more data leads to better accuracy of neural networks. We can add more images to the dataset and again train the model to improve its accuracy. Images that were uploaded by patients can also be used for training. Our system can be deployed on the portal of hospitals and can be used to assist oncologists to detect cancer.

With given annotations of the cancer cells, we can train new models which will not only classify if the image contains cancer cell but it will also give the position of these cancer cells in the image. Similar models can also be trained to detect different types of cancers if the proper dataset is provided.

Our proposed system is based on the web, because of which it requires internet connectivity to connect to the server where the models are predicting the results. An offline application interface can be created to use our system in the case of the unavailability of an internet connection. Such a system would be very useful in rural areas.

Bibliography

[1] Machines Will be Able to Diagnose Cancer. Accessed: Mar. 10, 2018. Available: http://sciencenordic.com/machines-will-be-able-diagnose-cancer.

[2] Convolutional Neural Networks for Image Classification and Captioning, Available: https://web.stanford.edu/class/cs231a/prev-projects-2016/example-paper.pdf.

[3] EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks Available: https://arxiv.org/abs/1905.11946

[4] Recent Advances in Convolutional Neural Networks, Available: https://arxiv.org/pdf/1512.07108.pdf.

[5] @misctensorflow2015-whitepaper, title= TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, url=https://www.tensorflow.org/, note=Software available from tensorflow.org,

Acknowledgement

We have great pleasure in presenting the report on **Histopathologic Cancer Detection** We take this opportunity to express our sincere thanks towards our guide **Prof. Pravin Prakash Adivarekar** Department of Computer Engineering, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin H. Malve** Head of Department,Comp. Engineering, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof.Amol Kalugade** BE project co-ordinator, Department of Comp. Engineering, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1: Ketan Anil Muddalkar
Student ID1: 16102022

Student Name2: Rohan Prashant Dhere
Student ID2: 15102057