ADVANCED ACADEMIC CENTER

A CENTER FOR INTER-DISCIPLINARY RESEARCH
2020-21

TITLE

# “LUNG CANCER DETECTION”

SUPERVISED BY

SANDHYA REYYA

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS

# Advanced Academic Center

**( A Center For Inter-Disciplinary Research )**

This is to certify that the project titled

**“LUNG CANCER DETECTION”**

is a bonafide work carried out by the following students in partial fulfilment of the requirements for Advanced Academic Center intern, submitted to the chair, AAC during the academic year 2020-21.

| NAME | ROLL NO. | BRANCH |
|---|---|---|
| YADAVALLI VIKAS | 20241A12C0 | IT |
| KAVYA CHATTUMALA | 20241A0306 | MECH |
| K KUSHI REDDY | 20241A0583 | CSE |

| NAME | ROLL N0 | BRANCH |
|---|---|---|
| KANUMURI SHREYA VARMA | 20241A0528 | CSE |
| S.V.S KARTHIK | 20241A04N5 | ECE |

This work was not submitted or published earlier for any study

Dr/Ms./Mr.

| Project Supervisor | Dr.B.R.K.Reddy<br>Program Coordinator | Dr.Ramamurthy Suri<br>Associate Dean,AAC |
|---|---|---|

ADVANCED ACADEMIC CENTER

# ACKNOWLEDGEMENTS

We express our deep sense of gratitude to our respected Director, Gokaraju Rangaraju Institute of Engineering and Technology, for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we extend our appreciation to our respected Principal, for permitting us to carry out this project.

We are thankful to the Associate Dean, Advanced Academic Centre, for providing us an appropriate environment required for the project completion.

We are grateful to our project supervisor who spared valuable time to influence us with their novel insights.

We are indebted to all the above mentioned people without whom we would not have concluded the project.

# ABSTRACT

Lung cancer is one of the dangerous and life taking disease in the world. However, early diagnosis and treatment can save life. Although, CT scan imaging is best imaging technique in medical field, it is difficult for doctors to interpret and identify the cancer from CT scan images. Therefore, computer aided diagnosis can be helpful for doctors to identify the cancerous cells accurately. Many computers aided techniques using image processing and machine learning has been researched and implemented. The main aim of this research is to evaluate the various computer-aided techniques, analysing the current best technique and finding out their limitation and drawbacks and finally proposing the new model with improvements in the current best model. The method used was that lung cancer detection techniques were sorted and listed on the basis of their detection accuracy. The techniques were analysed on each step and overall limitation, drawbacks were pointed out. It is found that some has low accuracy and some has higher accuracy but not nearer to 100%. Therefore, our research targets to increase the accuracy towards 100%.

# INTRODUCTION

The project aim is to detect the lung cancer cells using deep learning. Deep learning is the subfield of machine learning and neural networks make up the backbone of deep learning algorithms. A neural network is a series of algorithms that endeavours to recognise underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurones, either organic or artificial in nature.

A key feature of neural networks is an iterative learning process in which records (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all, cases are presented, the process is often repeated. During this learning phase, the network trains by adjusting the weights to predict the correct class label of input samples. The advantages of neural networks include their high tolerance to noisy data, as well as their ability to classify patterns on which they have not been trained. The most popular neural network algorithm is the back propagation algorithm.

Once a network has been structured for a particular application, that network is ready to be trained. To start this process, the initial weights (described in the next section) are chosen randomly. Then the training (learning) begins.

The network processes the records in the “training set” one at a time, using the weights and functions in the hidden layers, then compares the resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights for application to the next record.

This process occurs repeatedly as the weights are tweaked. During the training of a network, the same set of data is processed many times as the connection weights are continually refined. Today, neural networks are used for solving many business problems such as sales forecasting, customer research, data validation, and risk management. Neural networks are revolutionising business and everyday life, bringing us to the next level in artificial intelligence (AI). By emulating the way interconnected brain cells function, NN-enabled machines (including the smartphones and computers that we use on a daily basis) are now trained to learn, recognise patterns, and make predictions in a humanoid fashion as well as solve problems in every business sector.

Cancer is a group of diseases characterised by the uncontrolled growth and spread of abnormal cells. If the spread is not controlled, it can result in death. Lung cancer was the most common cancer in worldwide, contributing 2,093,876 of the total number of new cases diagnosed in 2018. The incidence rate has been declining since the mid- 1980s in men, but only since the mid-2000s in women, because of gender differences in historical patterns of smoking uptake and cessation. From 2005 to 2015, lung cancer incidence rates decreased by 2.5% per year in men and 1.2% per year in women. Symptoms do not usually occur until the cancer is advanced, and may include persistent cough, sputum streaked with blood, chest pain, voice change, worsening shortness of breath, and recurrent pneumonia or bronchitis.

Cigarette smoking is by far the most important risk factor for lung cancer; 80% of lung cancer deaths in the US are still caused by smoking. Risk increases with both quantity and duration of smoking. Cigar and pipe smoking also increase risk. Exposure to radon gas released from soil and building materials is thought to be the second-leading cause of lung cancer in the US.

Other risk factors include occupational or environmental exposure to second hand smoke, asbestos (particularly among smokers), certain metals (chromium, cadmium, arsenic), some organic chemicals, radiation, air pollution, and diesel exhaust. Some specific occupational exposures that increase risk include rubber manufacturing, paving, roofing, painting, and chimney sweeping. Risk is also probably increased among people with a history of tuberculosis. Genetic susceptibility (e.g., family history) plays a role in the development of lung cancer, especially in those who develop the disease at a young age.

We can cure lung cancer, only if you identifying the early stage. So here, we use machine learning algorithms to detect the lung cancer. This can be made faster and more accurate. In this study we propose machine learning strategies to improve cancer characterisation.

## Project workflow

The following is the workflow followed to complete the project.

1.Data gathering

2.Pre-processing of data

3.Model research

4.Training and testing the model

5.Analysis

## Data gathering

Collecting data for training the ML model is the basic step in the machine learning pipeline. The predictions made by ML systems can only be as good as the data on which they have been trained. Following are some of the problems that can arise in data collection:

- Inaccurate data. The collected data could be unrelated to the problem statement.
- Missing data. Sub-data could be missing. That could take the form of empty values in columns or missing images for some class of prediction.
- Data imbalance. Some classes or categories in the data may have a disproportionately high or low number of corresponding samples. As a result, they risk being under-represented in the model.
- Data bias. Depending on how the data, subjects and labels themselves are chosen, the model could propagate inherent biases on gender, politics, age or region, for example. Data bias is difficult to detect and remove.

## Pre-processing of data

Real-world raw data and images are often incomplete, inconsistent and lacking in certain behaviours or trends. They are also likely to contain many errors. So, once collected, they are pre-processed into a format the machine learning algorithm can use for the model.

Most of the real-world data is messy, some of these
types of data are:

1. Missing data: Missing data can be found when it is not continuously created or due to technical issues in the application (IOT system).

2. Noisy data: This type of data is also called outliners, this can occur due to human errors (human manually gathering the data) or some technical problem of the device at the time of collection of data.

3. Inconsistent data: This type of data might be collected due to human errors (mistakes with the name or values) or duplication of data.

These are some of the basic pre — processing techniques that can be used to convert raw data.

1. Conversion of data: As we know that Machine Learning models can only handle numeric features, hence categorical and ordinal data must be somehow converted into numeric features.

2. Ignoring the missing values: Whenever we encounter missing data in the data set then we can remove the row or column of data depending on our need. This method is known to be efficient but it shouldn’t be performed if there are a lot of missing values in the dataset.

3. Filling the missing values: Whenever we encounter missing data in the data set then we can fill the missing data manually, most commonly the mean, median or highest frequency value is used.

4. Machine learning: If we have some missing data then we can predict what data shall be present at the empty position by using the existing data.

5. Outliers detection: There are some error data that might be present in our data set that deviates drastically from other observations in a data set.

## Model research

A machine learning model is a file that has been trained to recognize certain types of patterns. You train a model over a set of data, providing it an algorithm that it can use to reason over and learn from those data.
Once you have trained the model, you can use it to reason over data that it hasn't seen before, and make predictions about those data. For example, let's say you want to build an application that can recognize a user's emotions based on their facial expressions. You can train a model by providing it with images of faces that are each tagged with a certain emotion, and then you can use that model in an application that can recognize any user's emotion.

## Classification

Classification problem is when the target variable is categorical (i.e. the output could be classified into classes — it belongs to either Class A or B or something else). A classification problem is when the output variable is a category, such as “red” or “blue”, “disease” or “no disease” or “spam” or “not spam”.
These some most used classification algorithms.

- K-Nearest Neighbour
- Naive Baye
- Decision Trees/Random Forest
- Support Vector Machine
- Logistic Regression

Regression:

While a Regression problem is when the target variable is continuous (i.e. the output is numeric).
These some most used regression algorithms.

- Linear Regression
- Support Vector Regression
- Decision Tress/Random Forest
- Gaussian Progresses Regression
- Ensemble Methods

Unsupervised learning:

In unsupervised learning, an AI system is presented with unlabelled, uncategorized data and the system's algorithms act on the data without prior training. The output is dependent upon the coded algorithms. Subjecting a system to unsupervised learning is one way of testing AI. The unsupervised learning is categorized into 2 other categories which are "Clustering" and "Association".
Clustering:

A set of inputs is to be divided into groups. Unlike in classification, the groups are not known beforehand, making this typically an unsupervised task.
Methods used for clustering are:

- Gaussian mixtures
- K-Means Clustering
- Boosting
- Hierarchical Clustering
- K-Means Clustering
- Spectral Clustering

## Training and testing the model

For training a model we initially split the model into 3 three sections which are 'Training data', 'Validation data' and 'Testing data'. We train the classifier using 'training data set', tune the parameters using 'validation set' and then test the performance of your classifier on unseen 'test data set'. An important point to note is that during training the classifier only the training and/or validation set is available. The test data set must not be used during training the classifier. The test set will only be available during testing the classifier.
Training set: The training set is the material through which the computer learns how to process information. Machine learning uses algorithms to perform the training part. A set of data used for learning, that is to fit the parameters of the classifier.
Validation set: Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. A set of unseen data is used from the training data to tune the parameters of a classifier.

Test set: A set of unseen data used only to assess the performance of a fully-specified classifier.
In a data set, a training set is implemented to build up a model, while a test (or validation) set is to validate the model built. Data points in the training set are excluded from the test (validation) set. Usually, a data set is divided into a training set, a validation set (some people use ‘test set’ instead) in each iteration, or divided into a training set, a validation set and a test set in each iteration. The model is trained we can use the same trained model to predict using the testing data i.e., the unseen data. Once this is done, we can develop a confusion matrix, this tells us how well our model is trained.

## Analysis

Model Analysis is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. To improve the model, we might tune the hyper-parameters of the model and try to improve the accuracy and also looking at the confusion matrix to try to increase the number of true positives and true negatives.

# CODE

```
#Importing required Libraries
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from tensorflow.keras.applications.imagenet_utils import preprocess_input
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import os

#Importing data initializing
DIRECTORY = r’C: \Users\vikas\Desktop\ML\train'
CATEGORIES = ["WITHOUT_CANCER", "WITH_CANCER"]



data = []
labels = []
```

```
for category in CATEGORIES:
    path = os.path.join(DIRECTORY, category)
    for img in os.listdir(path):
        img_path = os.path.join(path, img)
        image = load_img(img_path, target_size=(224, 224))
        image = img_to_array(image)
        image = preprocess_input(image)
        data.append(image)
        labels.append(category)


#Performing label binarizer
lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)
data = np.array(data,dtype = 'float32')
labels = np.array(labels)
#Splitting of data
(trainX , testX, trainY, testY) = train_test_split(data, labels,test_size=0.20, stratify=labels, random_state=0)
#Constructing the training image generator for data augmentation
aug = ImageDataGenerator(
            rotation_range=20,
            zoom_range=0.15,
            width_shift_range=0.2,
            height_shift_range=0.2,
            shear_range=0.15,
            horizontal_flip=True,
            fill_mode="nearest")
```

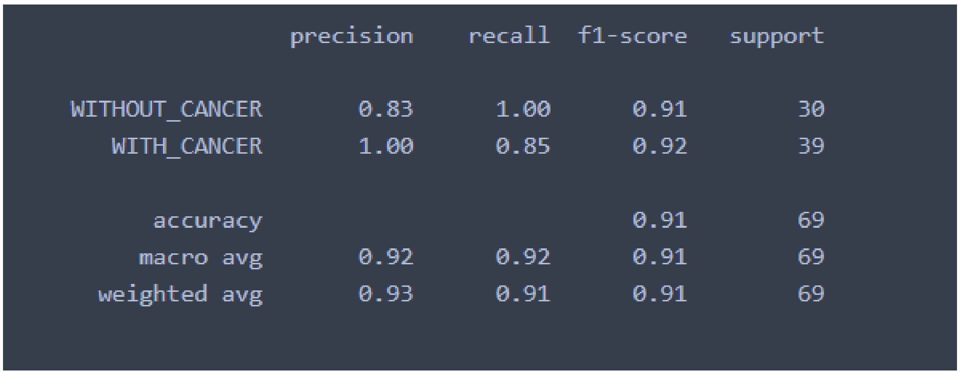
```
#Model training
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D
from tensorflow.keras.layers import MaxPooling2D
detector =  Sequential()
detector.add(Conv2D(64,(3,3),input_shape = (224,224,3), activation='relu'))
detector.add(Conv2D(64,(3,3),activation='relu'))
detector.add(MaxPooling2D(pool_size = (2,2)))
detector.add(Flatten())
detector.add(Dense(units=128,activation='relu'))
detector.add(Dense(units=2,activation='softmax'))
detector.compile(optimizer='adam',loss='binary_crossentropy',metrics=['accuracy'])

H = detector.fit(aug.flow(trainX, trainY,batch_size = 16),steps_per_epoch = len(trainX)//16,validation_data=(testX, testY),validation_steps=len(testX),epochs=10)
```

```
#Plot the accuracy
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, 10), H.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, 10), H.history["val_accuracy"], label="val_acc")
plt.title("Training Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Accuracy")
plt.legend(loc="upper right")
```

```
#Make predictions on testing set
predIdxs = detector.predict(testX, batch_size=16)
predIdxs = np.argmax(predIdxs, axis=1)
print(classification_report(testY.argmax(axis=1), predIdxs,target_names=lb.classes_))
```

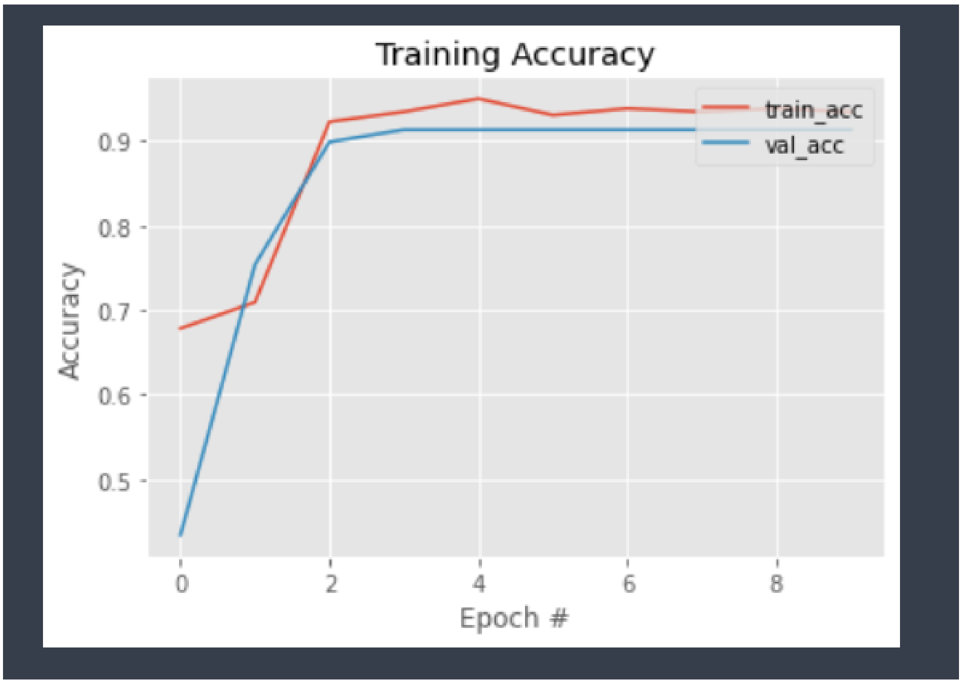
```
#Model Save
detector.save(r'C:\Users\vikas\Desktop\ML\LCD1')
```

#Classification report



```
                precision    recall  f1-score   support

WITHOUT_CANCER       0.83      1.00      0.91        30
   WITH_CANCER       1.00      0.85      0.92        39

      accuracy                           0.91        69
     macro avg       0.92      0.92      0.91        69
  weighted avg       0.93      0.91      0.91        69
```

#Accuracy vs Epoch graph



#Predicting model with image as input

import gradio as gr

import tensorflow as tf

import numpy as np

import requests

from tensorflow import keras

from tensorflow.keras.preprocessing.image import img_to_array

from tensorflow.keras.preprocessing.image import load_img

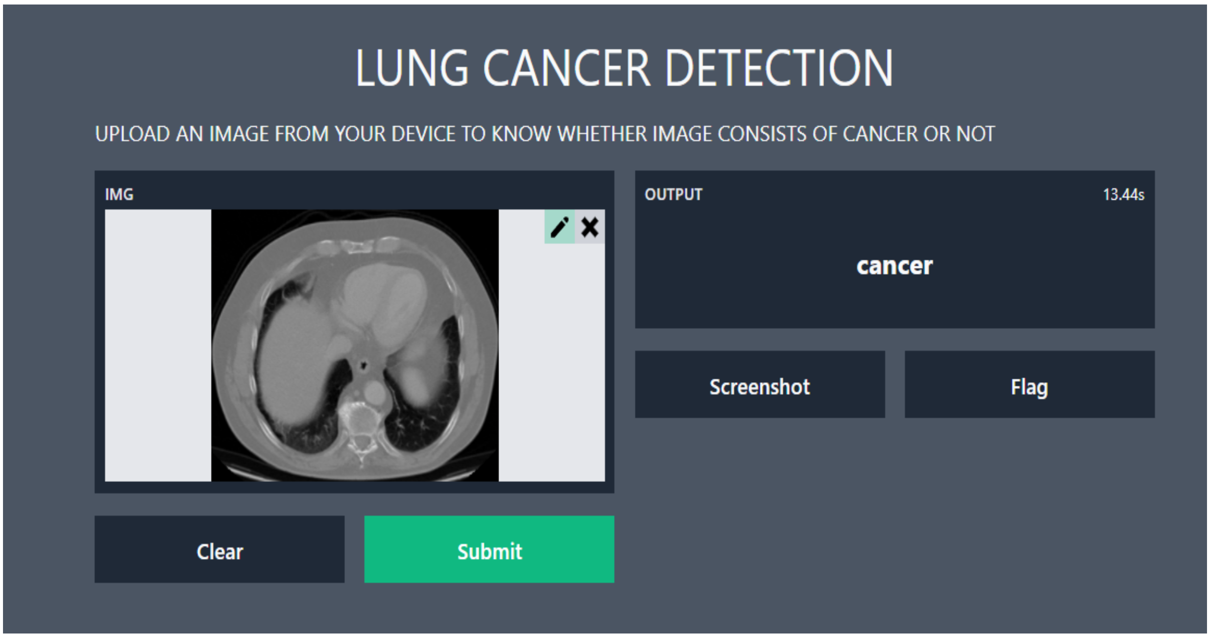
import cv2

from tensorflow.keras.preprocessing import image

```
def lcd(img):
    model = tf.keras.models.load_model("C:/Users/vikas/Desktop/ML/LCD1")
    img =cv2.resize(img,(224,224),interpolation=cv2.INTER_AREA)
    img = image.img_to_array(img)
    img = np.expand_dims(img, axis = 0)
    result = model.predict(img)
    if((result[0][0]==1) and (result[0][1]==0)):
         return("no cancer");
    else:
        return("cancer");
gr.Interface(fn=lcd, inputs="image", outputs="label",theme="darkgrass",title="LUNG CANCER DETECTION",description="UPLOAD AN IMAGE FROM YOUR DEVICE TO KNOW WHETHER IMAGE CONSISTS OF CANCER OR NOT").launch()
```

#Output

When image containing lung cancer is given as input.



LUNG CANCER DETECTION

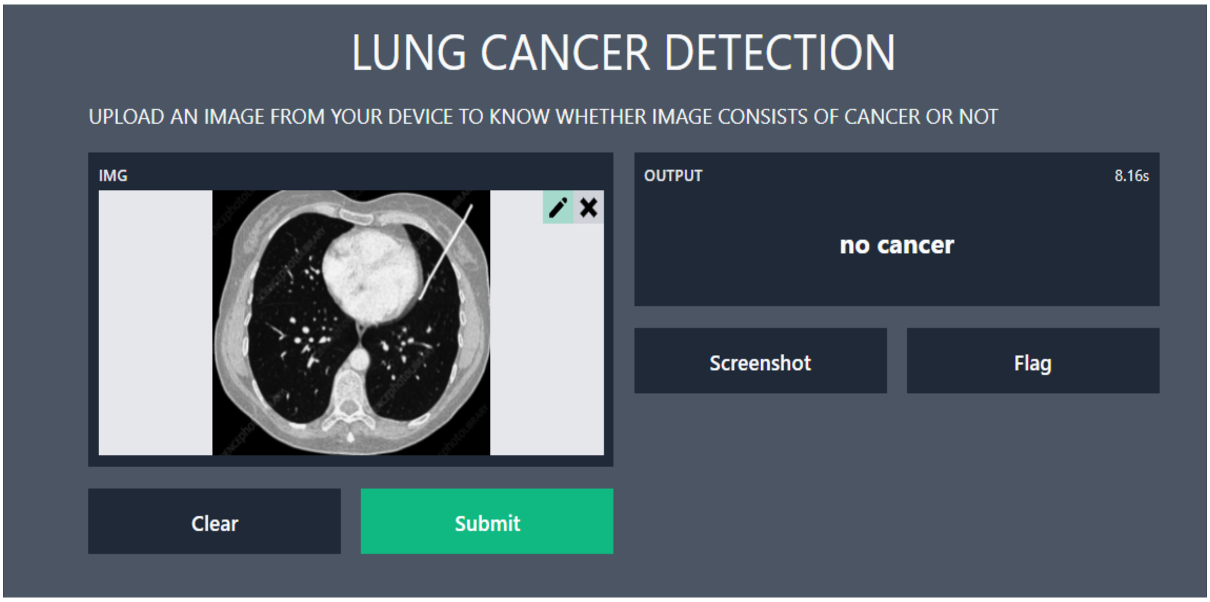
UPLOAD AN IMAGE FROM YOUR DEVICE TO KNOW WHETHER IMAGE CONSISTS OF CANCER OR NOT

IMG

OUTPUT 13.44s

cancer

When image not containing lung cancer is given as input.



LUNG CANCER DETECTION

UPLOAD AN IMAGE FROM YOUR DEVICE TO KNOW WHETHER IMAGE CONSISTS OF CANCER OR NOT

no cancer

# FUTURE DEVELOPMENTS

In addition to ongoing activities around data generation, access to computing power and resources, and addressing ethical, legal, and social issues, it is necessary to consider certain ways to support additional gaps in cancer-focused AI development.

Four important places that need to be concentrated for future developments are:

## -Identifying amenable research questions:

One area where AI/ML approaches are poised to make a significant impact is extracting information from multiscale and multimodal cancer datasets. Another area is digital pathology for cancer diagnosis. Digitizing slides and creating automated workflows for analysis has the potential to transform the field of pathology, and could improve the safety, accuracy, speed, and quality of pathology tests. For instance, NCI scientists developed a deep learning classifier for the detection of HPV-related precancers. The approach automates the evaluation of tissue slides stained for two markers, p16 and Ki-67, that are linked with HPV-related cervical carcinogenesis. The DL model had improved accuracy and efficiency compared with traditional Pap cytology, finding more precancers and reducing the number of false positives.

## -Customizing AI for cancer research:

There is a great need to develop AI/ML methodologies that are developed specifically for cancer research and care. Biological data, and especially cancer data, have features and limitations that differ significantly from that of other fields (including bias, noise, sample size, complexity, and patient privacy). In addition, it will be necessary to have access to ML models that incorporate knowledge of cancer biology and multiple data types. This will improve the accuracy of AIbased approaches for cancer-related applications.

## -Improve cancer care delivery:

Possible applications include clinical decision support activities such as identifying patient risk factors, treatment stratification, predicting risk of side effects and recurrence, and measuring and improving quality of care. However, while of exciting potential, the extraction of high-quality data for research uses from such real-world sources has proven complex and is a critical ongoing challenge to the field.

## -Implementing Ai based approaches:

Although the field of AI is fast expanding, there is a gap between AI/ML in research and clinical care. NCI has a responsibility to research to support the clinical implementation of AI/ML-based approaches.

# REFERENCES

- https://www.kaggle.com/mohamedhanyyy/chest-ctscan-images
- https://scikitlearn.org/stable/modules/generated/sklearn.preprocessing.LabelBinarizer.html
- https://www.upgrad.com/blog/basic-cnn-architecture/
- https://www.investopedia.com/terms/n/neuralnetwork.asp
- https://www.geeksforgeeks.org/python-pil-image-resize-method/
- https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529
- https://medium.com/analytics-vidhya/convolution-padding-stride-and-pooling-in-cnn-13dc1f3ada26
- https://holypython.com/how-to-batch-resize-multiple-images-in-python-via-pil-library/
- https://machinelearningknowledge.ai/keras-dense-layer-explained-for-beginners/
- https://medium.com/@PK_KwanG/cnn-step-2-flattening-50ee0af42e3e
- https://www.kite.com/python/answers/how-to-convert-an-image-to-an-array-in-python
- https://www.geeksforgeeks.org/how-to-convert-images-to-numpy-array/
- https://www.baeldung.com/cs/epoch-neural-networks
- https://deeplizard.com/learn/video/U4WB9p6ODjM
- https://www.analyticssteps.com/blogs/7-types-activation-functions-neural-network
- https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6
- https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/
- https://stackoverflow.com/questions/42516212/how-to-save-a-resized-image-in-python
- https://keras.io/api/layers/
- https://keras.io/api/preprocessing/image/
- https://www.w3resource.com/numpy/manipulation/reshape.php
- https://datascience.stackexchange.com/questions/29719/how-to-set-batch-size-steps-per-epoch-and-validation-steps
- https://www.sciencedirect.com/science/article/pii/S0304419X21000706