PATHOLOGY IMAGE ANALYSIS FOR LUNG CANCER CLASSIFICATION USING IBM WATSON

1. INTRODUCTION

1.1 Overview

The main objective of this project is to detect whether the tumor present in a patient's lung is malignant or benign using Convolution Neural Network (CNN)

In a study published in Nature Medicine, researchers said that lung cancer caused an estimated 160,000 deaths in 2018, making it the most common cause of cancer death in the US. Lung cancer screenings that use low-dose tomography have been shown to reduce mortality by 20-43 percent, but there are still challenges that result in unclear diagnoses, subsequent unnecessary procedures, and high costs. Radiologists also usually have to look through dozens of 2D images within a single CT scan, and cancer can be hard to spot. Deep learning can offer a viable solution to these problems.

.

1.2 Purpose

In healthcare, AI is used to improve clinical outcomes using innovative methods implicated in diagnostics and therapy particularly in oncology. There is mounting interest in the use of AI and ML to conduct complex calculation and assessing diagnostic images with minimal human intervention. This review will focus on AI and precision oncology in a clinical setting for cancer management by highlighting various applications of AI in oncology healthcare such as next generation sequencing (NGS), improvement in medical imaging, digital pathology and drug discovery..

2. LITERATURE SURVEY

2.1 Existing problem

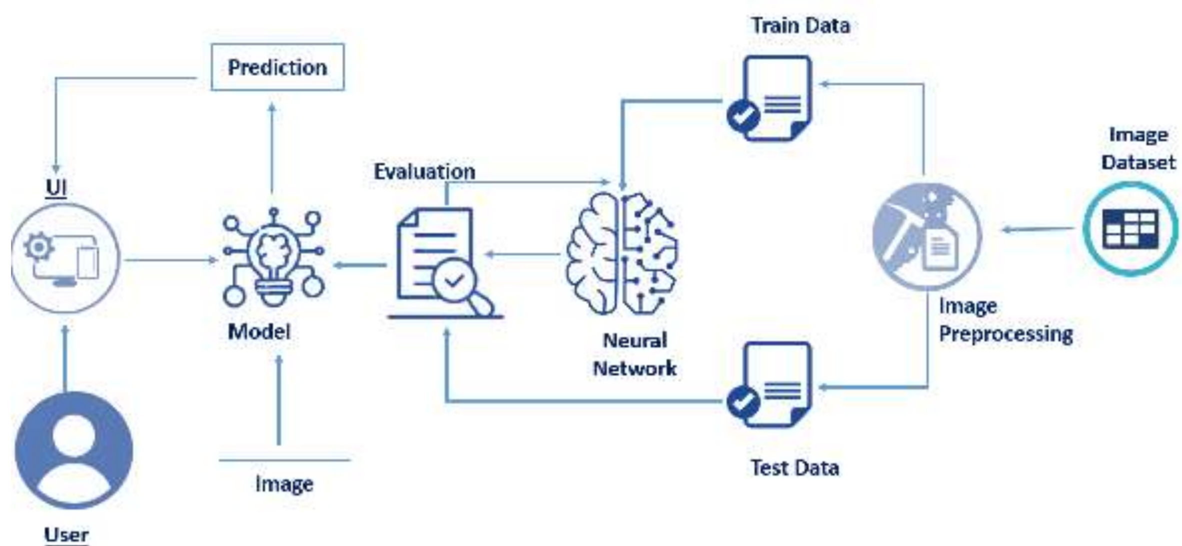
Pathology ananlysis for lung cancer classification using IBM Watson increases the role in medical field to easily detect cancer patients.. If the output is predicted more accurately,. In this project, we do energy prediction based on CT images of lungs of the patients.

2.2 Proposed solution

In medical image segmentation, the accuracy is foremost important, as it deals with human lives. It is highly crucial to eradicate the incidence of noise content and to improve the image quality before an examination . This part of work is known as preprocessing. In the preprocessing stage, noise removal and contrast enhancement are two primary steps. In the present study, the performance results of median, adaptive median, and average filters to isolate the presence of speckle noise have been compared. The coding for the same has been implemented using MATLAB. Furthermore, the image quality and visual appearance are improved by adaptive histogram equalization. The second stage of work is segmentation. This stage consists of applying five methods, namely, k-means, k-median, particle swarm optimization (PSO), inertia-weighted particle swarm optimization (IWPSO), and GCPSO. The tumor portion was extracted from the segmented results of the above-said five methods and compared with manual extraction. The results show that the GCPSO-based segmentation has more accuracy than the others. Figure 1 depicts the process of operation for the present study.

3. THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware / Software designing

Software Requirements:

- Anaconda Navigator
- Keras
- Flask

Hardware Requirements:

- Processor : Intel Core i3
- Hard Disk Space : Min 100 GB
- Ram : 8 GB
- Display : 14.1 "Color Monitor(LCD, CRT or LED)

  Clock Speed : 1.67 GHz

4. EXPERIMENTAL INVESTIGATIONS

Currently, the most hazardous disease that faced humanity's life and led to fatal death is referred to as Cancer . In other words, the abnormality of enhancing human cells and converting it into a tumor is known as Cancer. Among the various forms of cancer, the Lung-Cancer is enumerated as the riskiest one when compared to the other types around the globe. According to the doctors' view, the most essential factor that causes Lung-Cancer is tobacco and the survival rate of patients will increase if the cancerous tumor detects in its early age [3]. Among genders, the male is more faced to Lung-Cancer than female due to the higher ratio of smoking [4]. For instances, according to a report which showed that among (116,470) males and (109,690) females,(87,750) males and (72,590) females were died because of Lung-Cancer. In the World, merely one from five deaths is due to smoking and utilizing tobacco..

5.PROJECT FLOW

- User interacts with User interface to upload image
- Uploaded image is analyzed by the model which is integrated
- Once model analyses the uploaded image, the prediction is showcased on the UI To accomplish this, we have to complete all the activities and tasks listed below

- Data Collection.
    - Collect the dataset or Create the dataset
- Data Preprocessing.
    - Import the ImageDataGenerator library
    - Configure ImageDataGenerator class
    - Apply ImageDataGenerator functionality to Trainset and Testset
- Model Building
    - Import the model building Libraries
    - Initializing the model
    - Adding Input Layer
    - Adding Hidden Layer
    - Adding Output Layer
    - Configure the Learning Process
    - Training and testing the model
    - Optimize the Model
    - Save the Model
- Application Building
    - Create an HTML file
    - Build Python Code

6. RESULT


Lung Cancer Detection In Radiology Images
Using technology to save lives!
Leveraging the power of
AI and Deep Learning
to increase early detection of cancer!
Type here to search


Detect
Introduction
Lung cancer is one of the most killerdiseases in the developing countries and the detection of the cancer at the early stage is a challenge. Analysis and cure of lung malignancy have been one of the greatest difficulties faced by humans over the most recent couple of decades. Early identification of tumor would facilitate in sparing a huge number of lives over the globe consistently.
This project presents lung cancer detection based on chest CT images using CNN. In the first stage, lung regions are extracted from the CT images. The extracted images are used to train the CNN architecture. Then, CNN is used to test the patient images. The main objective of this study is to detect whether the tumor present in a patient's lung is malignant or benign. A web application is built which gives the feasibility of uploading a CT scan image. This app is integrated with a model. The model detects the tumor and the prediction is shown on the UI.
Upload your image
Select
Top
Copyright © 2020 RSS Pvt Ltd
Type here to search


Detect
Introduction
Lung cancer is one of the most killerdiseases in the developing countries and the detection of the cancer at the early stage is a challenge. Analysis and cure of lung malignancy have been one of the greatest difficulties faced by humans over the most recent couple of decades. Early identification of tumor would facilitate in sparing a huge number of lives over the globe consistently.
This project presents lung cancer detection based on chest CT images using CNN. In the first stage, lung regions are extracted from the CT images. The extracted images are used to train the CNN architecture. Then, CNN is used to test the patient images. The main objective of this study is to detect whether the tumor present in a patient's lung is malignant or benign. A web application is built which gives the feasibility of uploading a CT scan image. This app is integrated with a model. The model detects the tumor and the prediction is shown on the UI.
Top
Upload your image
Select
Copyright © 2020 RSS Pvt Ltd
Type here to search


Detect
Introduction
Lung cancer is one of the most killerdiseases in the developing countries and the detection of the cancer at the early stage is a challenge. Analysis and cure of lung malignancy have been one of the greatest difficulties faced by humans over the most recent couple of decades. Early identification of tumor would facilitate in sparing a huge number of lives over the globe consistently.
This project presents lung cancer detection based on chest CT images using CNN. In the first stage, lung regions are extracted from the CT images. The extracted images are used to train the CNN architecture. Then, CNN is used to test the patient images. The main objective of this study is to detect whether the tumor present in a patient's lung is malignant or benign. A web application is built which gives the feasibility of uploading a CT scan image. This app is integrated with a model. The model detects the tumor and the prediction is shown on the UI.
Top
Upload your image
Select
Copyright © 2020 RSS Pvt Ltd
Type here to search


Detect
Introduction
Lung cancer is one of the most killerdiseases in the developing countries and the detection of the cancer at the early stage is a challenge. Analysis and cure of lung malignancy have been one of the greatest difficulties faced by humans over the most recent couple of decades. Early identification of tumor would facilitate in sparing a huge number of lives over the globe consistently.
This project presents lung cancer detection based on chest CT images using CNN. In the first stage, lung regions are extracted from the CT images. The extracted images are used to train the CNN architecture. Then, CNN is used to test the patient images. The main objective of this study is to detect whether the tumor present in a patient's lung is malignant or benign. A web application is built which gives the feasibility of uploading a CT scan image. This app is integrated with a model. The model detects the tumor and the prediction is shown on the UI.
Top
Upload your image
Select
Copyright © 2020 RSS Pvt Ltd


Detect
Introduction
Lung cancer is one of the most killerdiseases in the developing countries and the detection of the cancer at the early stage is a challenge. Analysis and cure of lung malignancy have been one of the greatest difficulties faced by humans over the most recent couple of decades. Early identification of tumor would facilitate in sparing a huge number of lives over the globe consistently.
This project presents lung cancer detection based on chest CT images using CNN. In the first stage, lung regions are extracted from the CT images. The extracted images are used to train the CNN architecture. Then, CNN is used to test the patient images. The main objective of this study is to detect whether the tumor present in a patient's lung is malignant or benign. A web application is built which gives the feasibility of uploading a CT scan image. This app is integrated with a model. The model detects the tumor and the prediction is shown on the UI.
Top
Upload your image
Select
Copyright © 2020 RSS Pvt Ltd

7. ADVANTAGES

Advantages:

To overcome the aforementioned challenges, various image processing and machine learning methods have been proposed and have so far achieved great progress. However, it is important to note the advantages of deep learning methods over non-deep-learning methods (also called shallow-learning methods).

Currently, convolutional neural networks (CNNs) are the most frequently used deep learning model for image data classification, including tumor detection in pathology images of breast cancer , renal cell carcinoma , prostate cancer , and head and neck cancer . Several forms of neural network have been derived from CNNs for image segmentation , including fully convolutional networks (FCNs) and mask-regional convolutional neural networks (mask-RCNNs) . Recurrent neural networks (RNNs), which are well known for modeling dynamic sequence behavior such as speech recognition, have also been explored in multi-label image classification and image segmentation . In additional to the aforementioned supervised deep learning models, autoencoder, an unsupervised deep learning model, has shown ability in analyzing pathology images through pre-training models , cell detection , and image feature extraction .

8. APPLICATIONS

- Lung cancer is the uncontrolled growth of abnormal cells that starts off in one or both images. People who smoke have the greatest risk of lung cancer. The overall 5 year survival rate for lung cancer combining all stages is roughly 15%. Early detection of lung cancer can increase the chance of survival among people. Lung cancer may be found by imaging tests such as chest computed tomography scan as it provides more detailed picture. To classify the stages of lung cancer, image processing technique is developed. In this work, new algorithm is developed using image processing technique to detect the cancer at early stage with more accuracy. Image processing involves the pre-processing, feature extraction and finally classification steps..

9. CONCLUSION

In this study, various optimization algorithms have been evaluated to detect the tumor. Medical images often need preprocessing before being subjected to statistical analysis. The adaptive median filter has better results than median and mean filters because the speckle suppression index and speckle and mean preservation index values are lower for the adaptive median filter. Comparing the five algorithms, the accuracy of the tumor extraction is improved in GCPSO with the highest accuracy of 95.8079%, and it obtained above 90% of precision in all the 20 images. It is more accurate when compared to the previous method which had an accuracy of 90% in 4 out of 10 datasets only. In future studies, the use of more number of optimization algorithms will be included to improve the accuracy.

10. FUTURE SCOPE

Our attempt would be to further improve the predictions using the PATHOLOGY IMAGE ANALYSIS FOR LUNG CANCER CLASSIFICATION model are powerful. In medical image segmentation, the accuracy is foremost important, as it deals with human lives. It is highly crucial to eradicate the incidence of noise content and to improve the image quality before an examination [4]. This part of work is known as preprocessing. In the preprocessing stage, noise removal and contrast enhancement are two primary steps. In the present study, the performance results of median, adaptive median, and average filters to isolate the presence of speckle noise have been compared

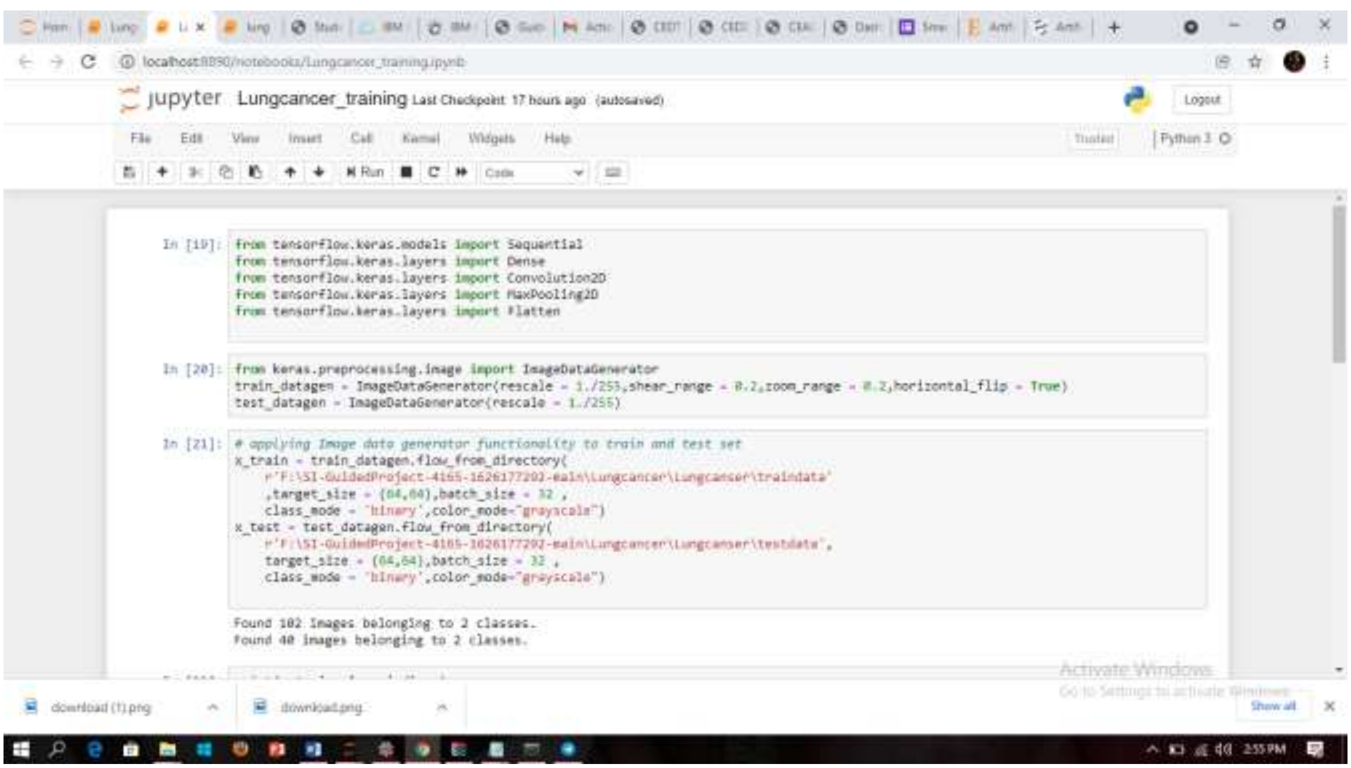
11. BIBILOGRAPHY

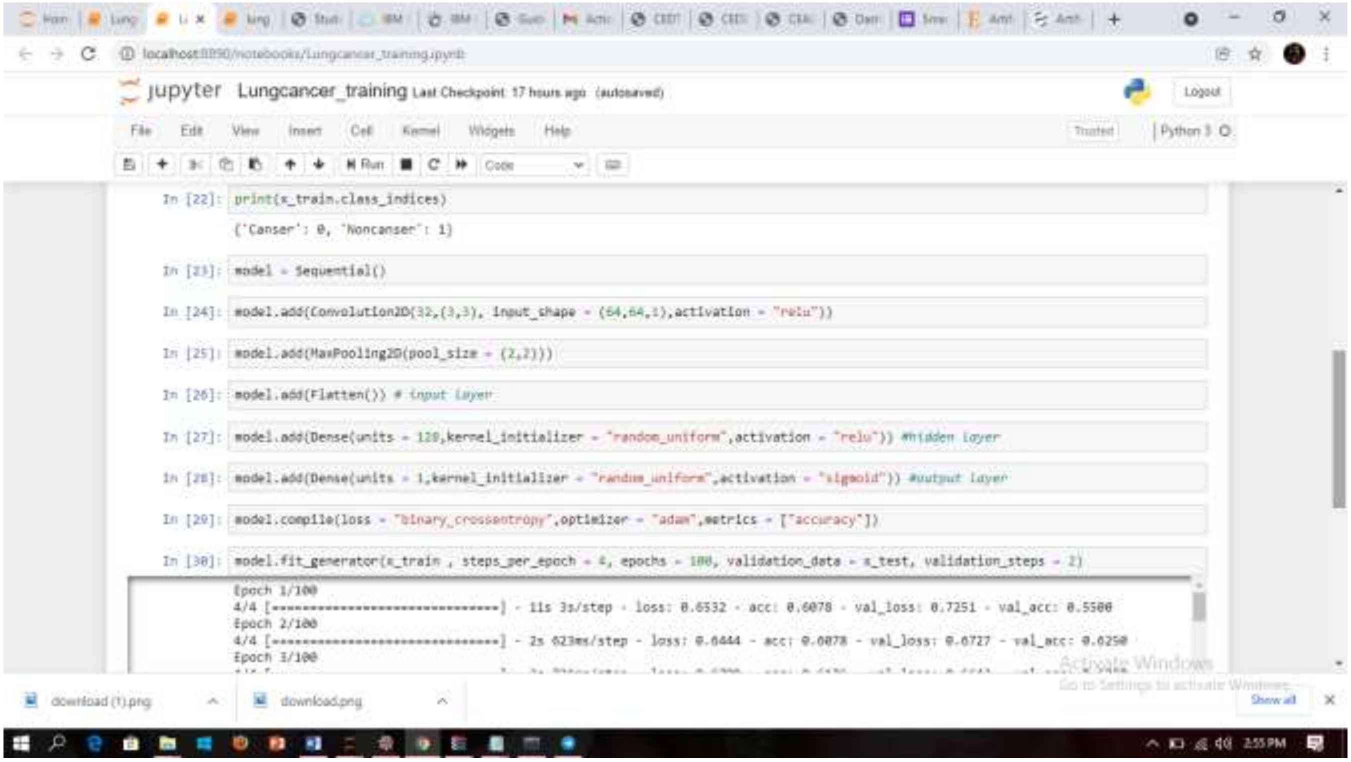
- A. Brindha, S. Indirani, and A. Srinivasan, "Lung cancer detection using SVM algorithm and optimization techniques," Journal of Chemical and Pharmaceutical Sciences, vol. 9, no. 4, 2016.View at: Google Scholar

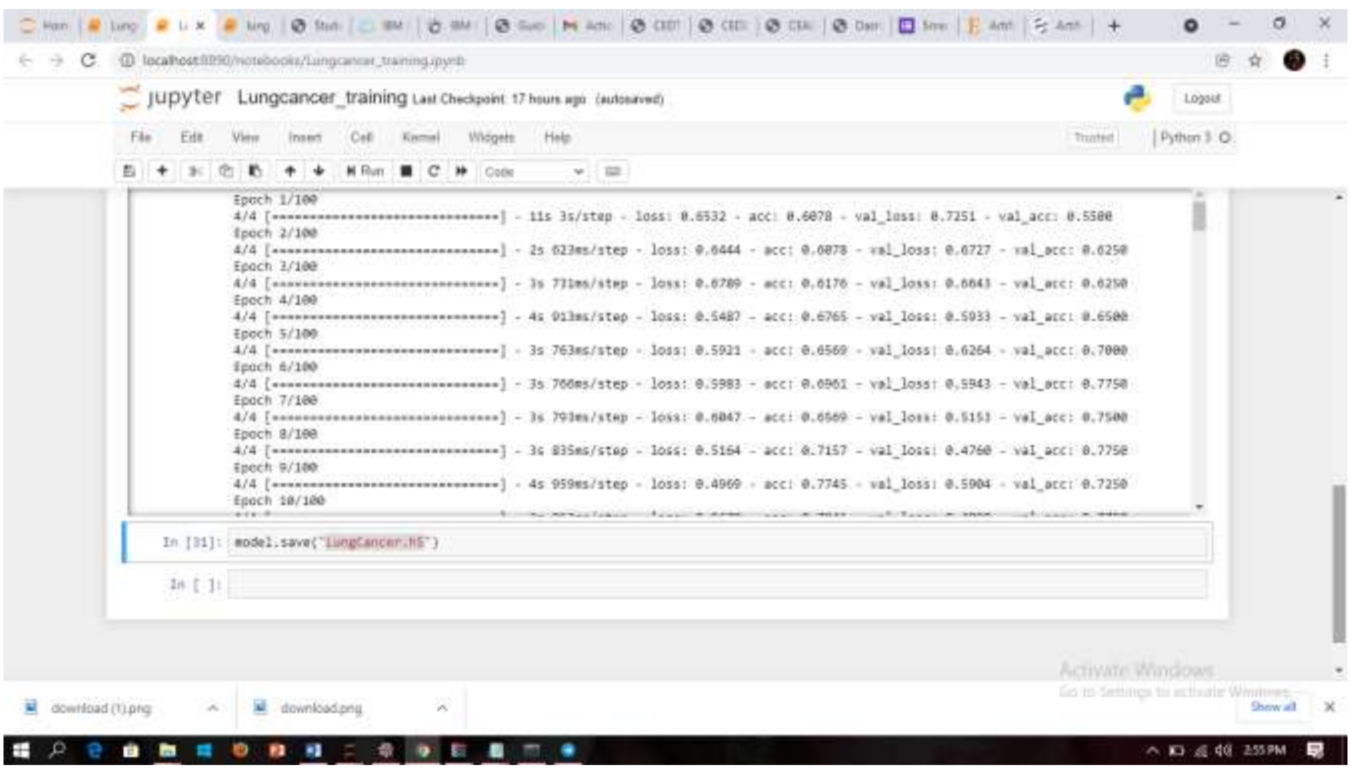
- M. Kurkure and A. Thakare, “Introducing automated system for lung cancer detection using Evolutionary Approach,” International Journal of Engineering and Computer Science, vol. 5, no. 5, pp. 16736–16739, 2016.View at: Google Scholar

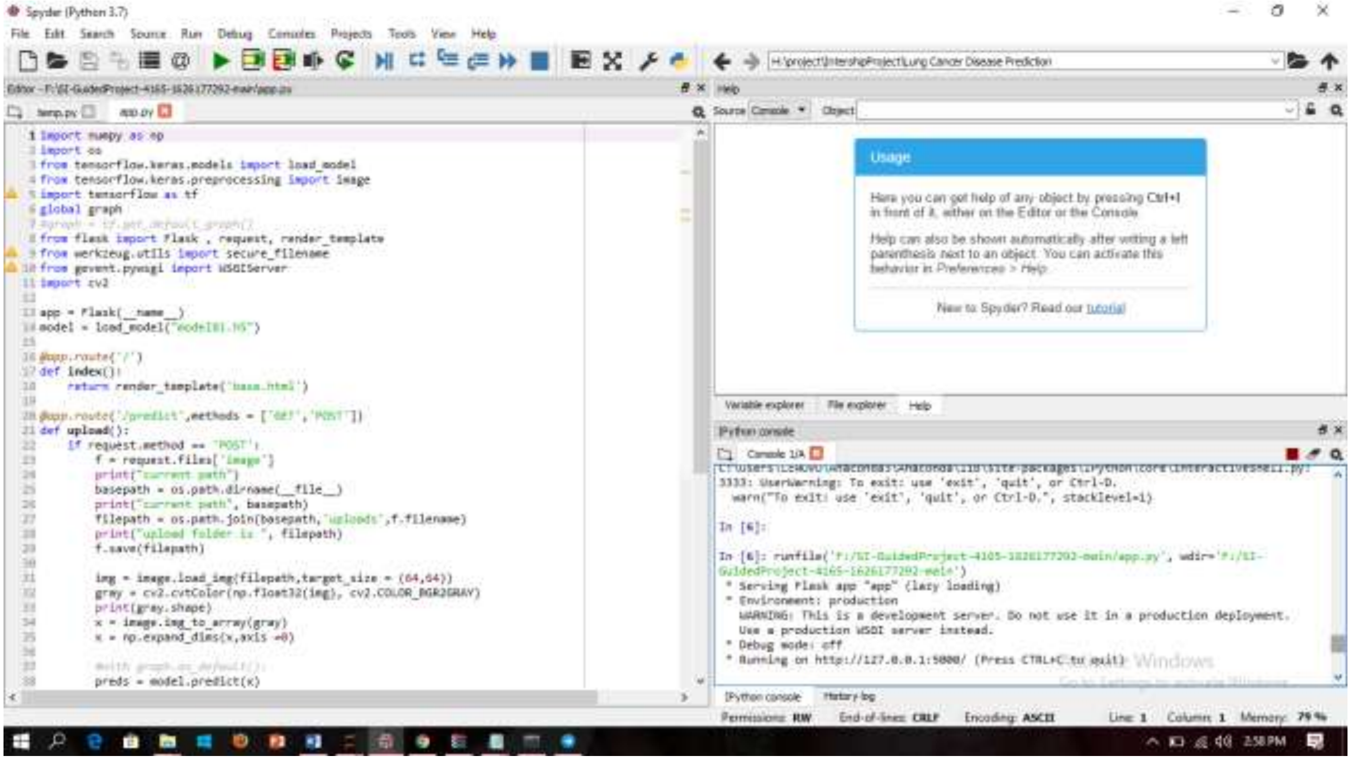
APPENDIX

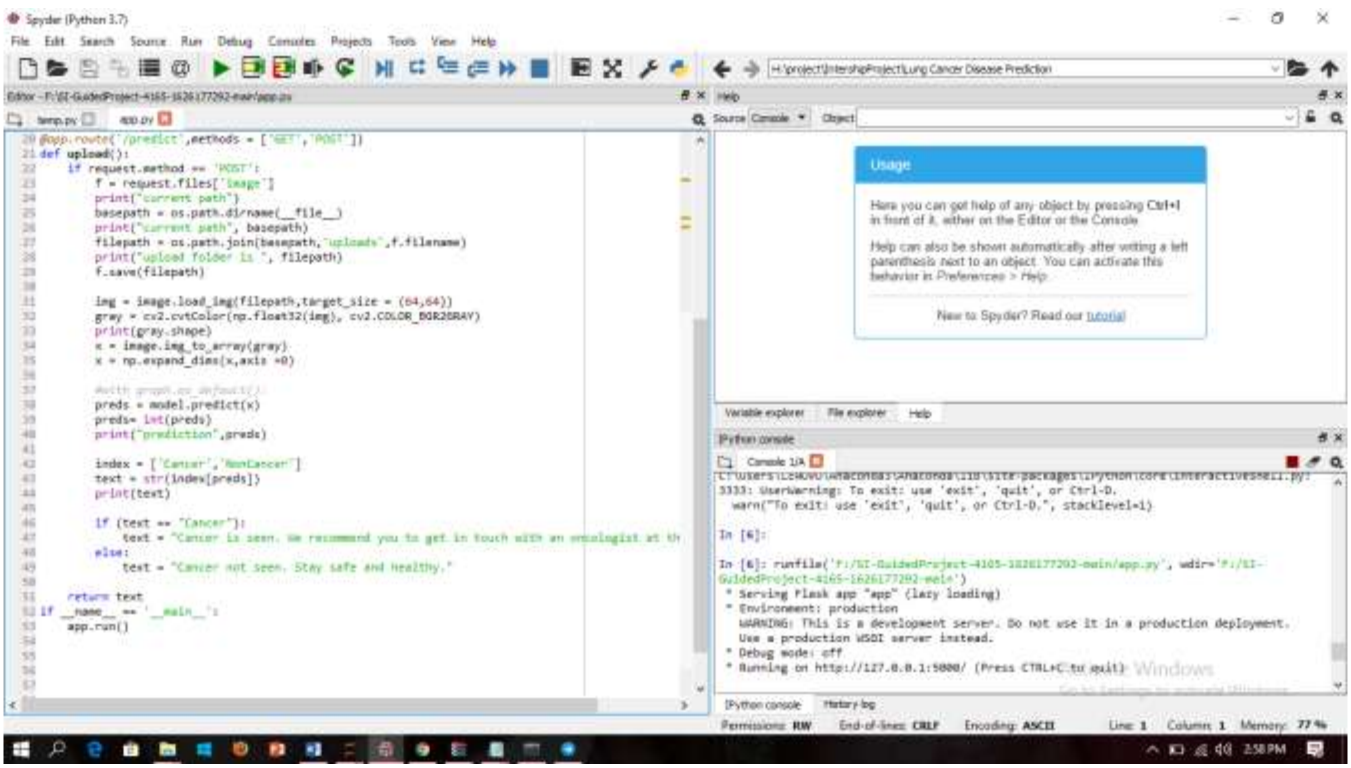
Source Code




jupyter Lungcancer_training Last Checkpoint: 17 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
In [22]: print(x_train.class_indices)
{'Canser': 0, 'Noncanser': 1}
In [23]: model = Sequential()
In [24]: model.add(Convolution2D(32,(3,3), input_shape = (64,64,1),activation = "relu"))
In [25]: model.add(MaxPooling2D(pool_size = (2,2)))
In [26]: model.add(Flatten()) # input layer
In [27]: model.add(Dense(units = 128,kernel_initializer = "random_uniform",activation = "relu")) #hidden layer
In [28]: model.add(Dense(units = 1,kernel_initializer = "random_uniform",activation = "sigmoid")) #output layer
In [29]: model.compile(loss = "binary_crossentropy",optimizer = "adam",metrics = ["accuracy"])
In [30]: model.fit_generator(x_train , steps_per_epoch = 4, epochs = 100, validation_data = x_test, validation_steps = 2)
Epoch 1/100
4/4 [==============================] - 11s 3s/step - loss: 0.6532 - acc: 0.6078 - val_loss: 0.7251 - val_acc: 0.5500
Epoch 2/100
4/4 [==============================] - 2s 623ms/step - loss: 0.6444 - acc: 0.6078 - val_loss: 0.6727 - val_acc: 0.6250
Epoch 3/100
download (1).png
download.png
Show all


jupyter Lungcancer_training Last Checkpoint: 17 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Epoch 1/100
4/4 [==============================] - 11s 3s/step - loss: 0.6532 - acc: 0.6078 - val_loss: 0.7251 - val_acc: 0.5500
Epoch 2/100
4/4 [==============================] - 2s 623ms/step - loss: 0.6444 - acc: 0.6078 - val_loss: 0.6727 - val_acc: 0.6250
Epoch 3/100
4/4 [==============================] - 3s 731ms/step - loss: 0.6789 - acc: 0.6176 - val_loss: 0.6643 - val_acc: 0.6250
Epoch 4/100
4/4 [==============================] - 4s 913ms/step - loss: 0.5487 - acc: 0.6765 - val_loss: 0.5933 - val_acc: 0.6500
Epoch 5/100
4/4 [==============================] - 3s 763ms/step - loss: 0.5921 - acc: 0.6569 - val_loss: 0.6264 - val_acc: 0.7000
Epoch 6/100
4/4 [==============================] - 3s 766ms/step - loss: 0.5983 - acc: 0.6961 - val_loss: 0.5943 - val_acc: 0.7750
Epoch 7/100
4/4 [==============================] - 3s 793ms/step - loss: 0.6047 - acc: 0.6569 - val_loss: 0.5153 - val_acc: 0.7500
Epoch 8/100
4/4 [==============================] - 3s 835ms/step - loss: 0.5164 - acc: 0.7157 - val_loss: 0.4760 - val_acc: 0.7750
Epoch 9/100
4/4 [==============================] - 4s 959ms/step - loss: 0.4969 - acc: 0.7745 - val_loss: 0.5904 - val_acc: 0.7250
Epoch 10/100
In [31]: model.save("LungCancer.h5")
In [ ]:
download (1).png
download.png
Show all


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
import numpy as np
import os
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import tensorflow as tf
global graph
#graph = tf.get_default_graph()
from flask import Flask , request, render_template
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
import cv2
app = Flask(__name__)
model = load_model("model181.h5")
@app.route('/')
def index():
return render_template('base.html')
@app.route('/predict',methods = ['GET','POST'])
def upload():
if request.method == 'POST':
f = request.files['image']
print("current path")
basepath = os.path.dirname(__file__)
print("current path", basepath)
filepath = os.path.join(basepath,'uploads',f.filename)
print("upload folder is ", filepath)
f.save(filepath)
img = image.load_img(filepath,target_size = (64,64))
gray = cv2.cvtColor(np.float32(img), cv2.COLOR_BGR2GRAY)
print(gray.shape)
x = image.img_to_array(gray)
x = np.expand_dims(x,axis =0)
#with graph.as_default():
preds = model.predict(x)
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Python console
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
@app.route('/predict',methods = ['GET','POST'])
def upload():
if request.method == 'POST':
f = request.files['image']
print("current path")
basepath = os.path.dirname(__file__)
print("current path", basepath)
filepath = os.path.join(basepath,'uploads',f.filename)
print("upload folder is ", filepath)
f.save(filepath)
img = image.load_img(filepath,target_size = (64,64))
gray = cv2.cvtColor(np.float32(img), cv2.COLOR_BGR2GRAY)
print(gray.shape)
x = image.img_to_array(gray)
x = np.expand_dims(x,axis =0)
#with graph.as_default():
preds = model.predict(x)
preds= int(preds)
print("prediction",preds)
index = ['Cancer','NonCancer']
text = str(index[preds])
print(text)
if (text == "Cancer"):
text = "Cancer is seen. We recommend you to get in touch with an oncologist at th
else:
text = "Cancer not seen. Stay safe and healthy."
return text
if __name__ == '__main__':
app.run()
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Python console
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)