B

Install App   ✕

ack your progress on the go!

← **Back**

Congrats Daniel! This project has been marked as completed.

Project Rating

★★★★☆

Teacher's Comment

"Good"

-

| Was this helpful? |

Community Link

| **Publish to Community** |

Edit Your Project

| **Last Submitted** |

Previous Submissions

| 28th Jan 2024 | **Open Link** |

| **Start Project** |

Submit Your Project

Learn how to submit your project ▶

| Paste your project URL |

| **Submit Project** |

Class Summary

This project is based on your last class PRO-C113

| **View Class Summary** |

PRO-C113: DAMAGED BUILDING DETECTION III `Completed`

In Class 113, You Learned To Compile Your Model For The Correct Image Class. In This Project, You'll Be Compiling The Previously Created Cnn Model And Testing The Model To Detect The Buildings Damaged By Hurricane Harvey In Texas.

Goal of the Project:

In Class 113, you learned to compile your model for the correct image class. In this project, you'll be compiling the previously created CNN model and testing the model to detect the buildings damaged by Hurricane Harvey in Texas.

Story:

Chris works in the weather department in Texas. He wants to calculate the total damage caused to buildings by Hurricane Harvey. For this, he needs the total number of buildings damaged by the hurricane. He has the data in the form of images. We have to help him segregate these images into the damaged buildings and not damaged buildings by the hurricane.

In Project 112, we had created a CNN model. In this project, you have to compile and fit the Convolutional Neural Network(CNN) model for training. Also, test the model which can predict the class of the image.

| Compile CNN model ▶ | Reduce Loss using optimizer ▶ | Fit and save the model ▶ | Test the model |
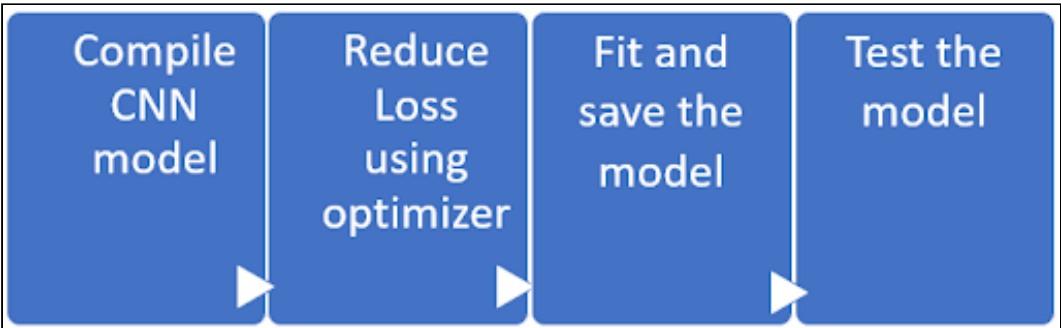
*This is just for your reference. We expect you to apply your own creativity to the project.

Getting Started:

1. Open the template link and click on **Open with Google Colaboratory.** (If it opens Google Colab then go to file>>>save a copy>>>start coding
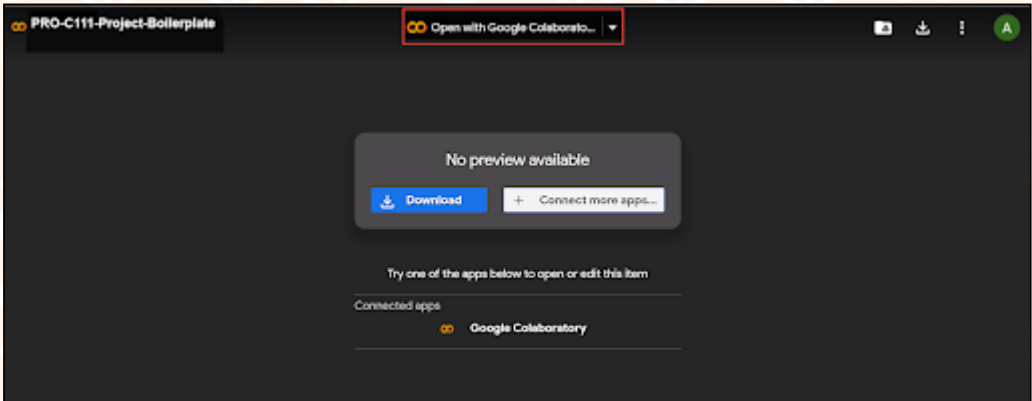
2. Clone the dataset from the link and run the initial code to display an image from the dataset.

3. Perform Image augmentation as the first step to process images before training the model.
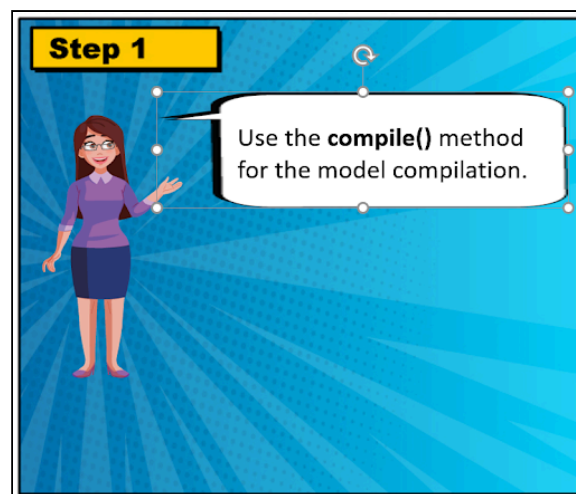
| 💬 Ask a doubt to your tea|
| ❓ HELP |

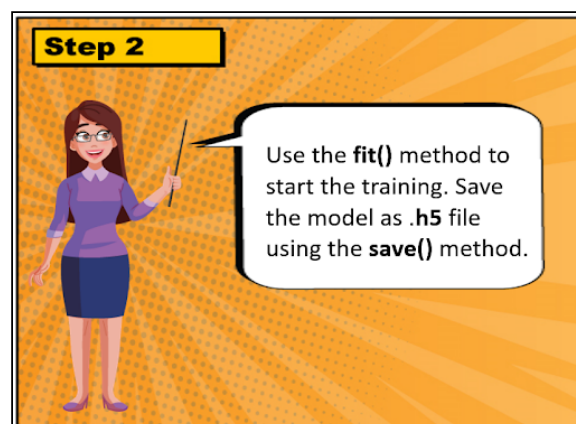4. Create a CNN model defining different layers to define a model.

5. We have performed these steps in Project 112. So you can continue compiling same model here.

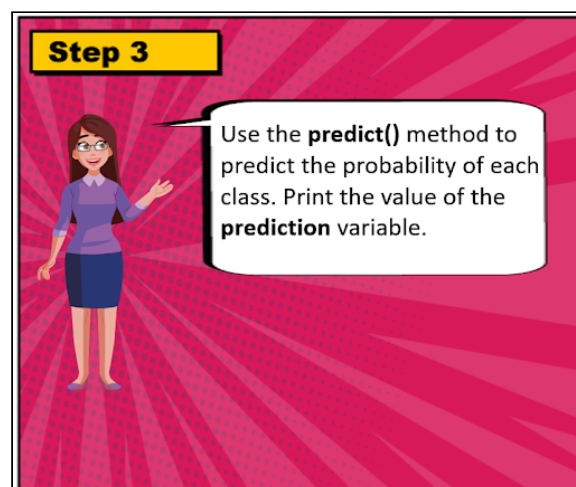## Specific Tasks to complete the Project:

Step 1

Use the **compile()** method for the model compilation.

```python
#Compile Model
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

Step 2

Use the **fit()** method to start the training. Save the model as **.h5** file using the **save()** method.

```python
history = model.fit(training_augmented_images, epochs=20,
                    validation_data = validation_augmented_images,
                    verbose=True)

model.save("Hurricane_damage.h5")
```

Step 3

Use the **predict()** method to predict the probability of each class. Print the value of the **prediction** variable.

Submitting the Project:

1. **SAVE** all the changes made to the project.
2. Click on **"Run"** once to check if it is working.
3. Rename the project to **Project 113**.
4. Click **Share**.

HELP
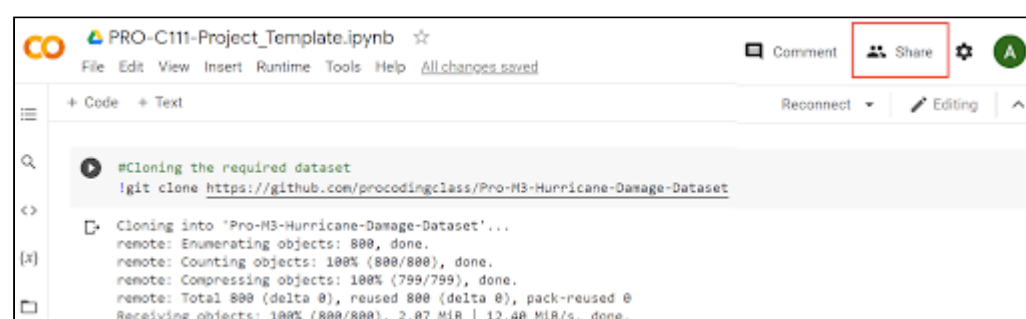
5. Click **Change** and choose the **'anyone with the link'** option.

6. Copy the link and submit it in the Student Dashboard Projects panel against the correct class number.

Hints:

1. Consider the following steps for predicting the class of image.
   1. Use the test image directory
   2. Add all the image files in directory
   3. Loop through nine image files
   4. Provide the exact path of the image and Load the image
   5. Convert the images to arrays and increase the dimensions
   6. Predict the class of unseen image
   7. Print and plot the image