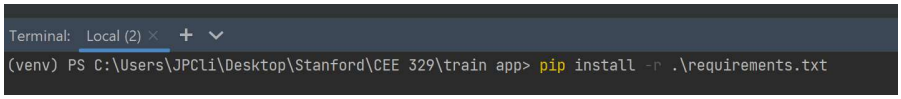
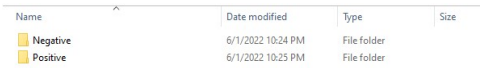
Train Model Page Instructions

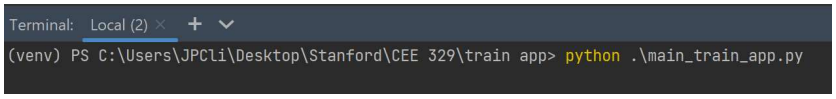
1. Install requirements.txt in Terminal:
   pip install -r .\requirements.txt

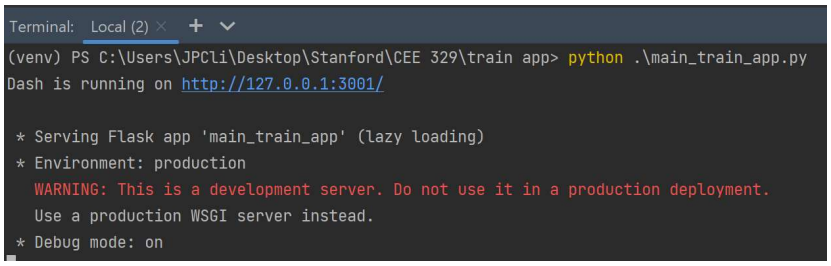


   And
   pip install “dash[diskcache]”
2. Your work folder should include:
   a) main_train_app.py
   b) split_dataset.py train_model.py evaluate_model.py
   c) dataset (the raw dataset is downloaded from
      https://data.mendeley.com/datasets/5y9wdsg2zt/1
      and rename it into ‘dataset’, including Negative and Positive)

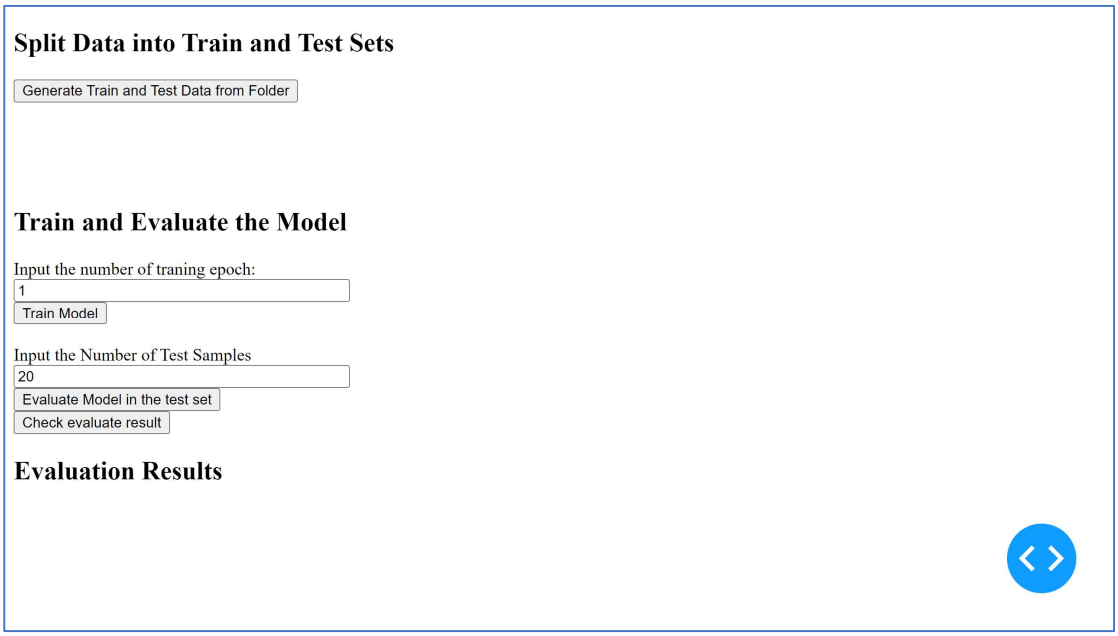


3. Run the file: main_train_app.py in Terminal
   python .\main_train_app.py

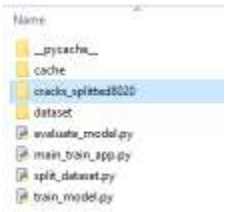


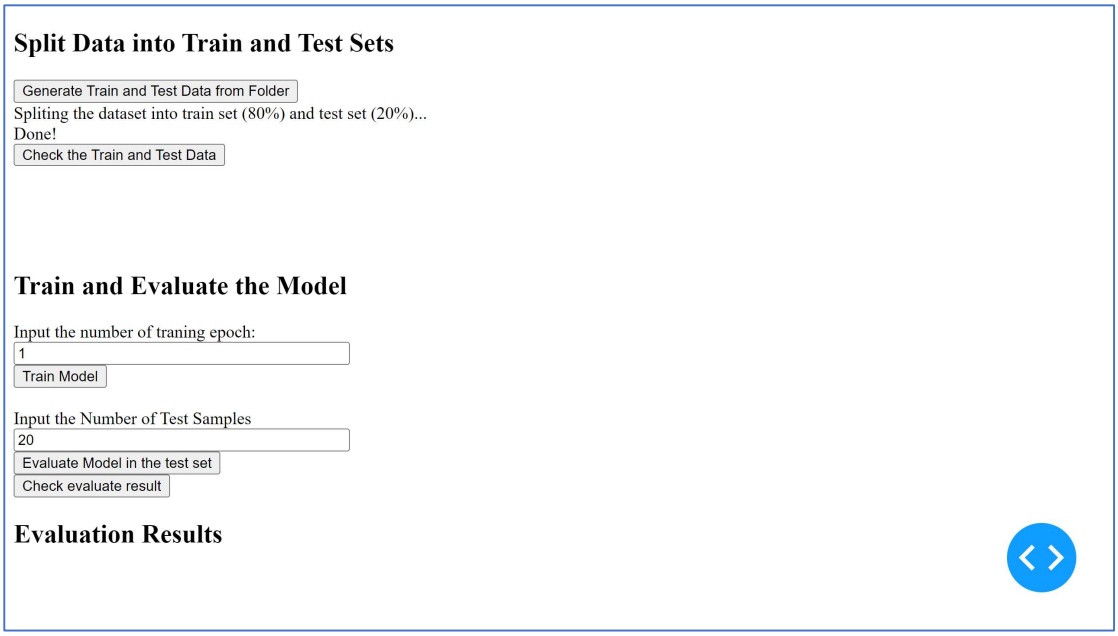


4. Open the link http://127.0.0.1:3001/ in a browser

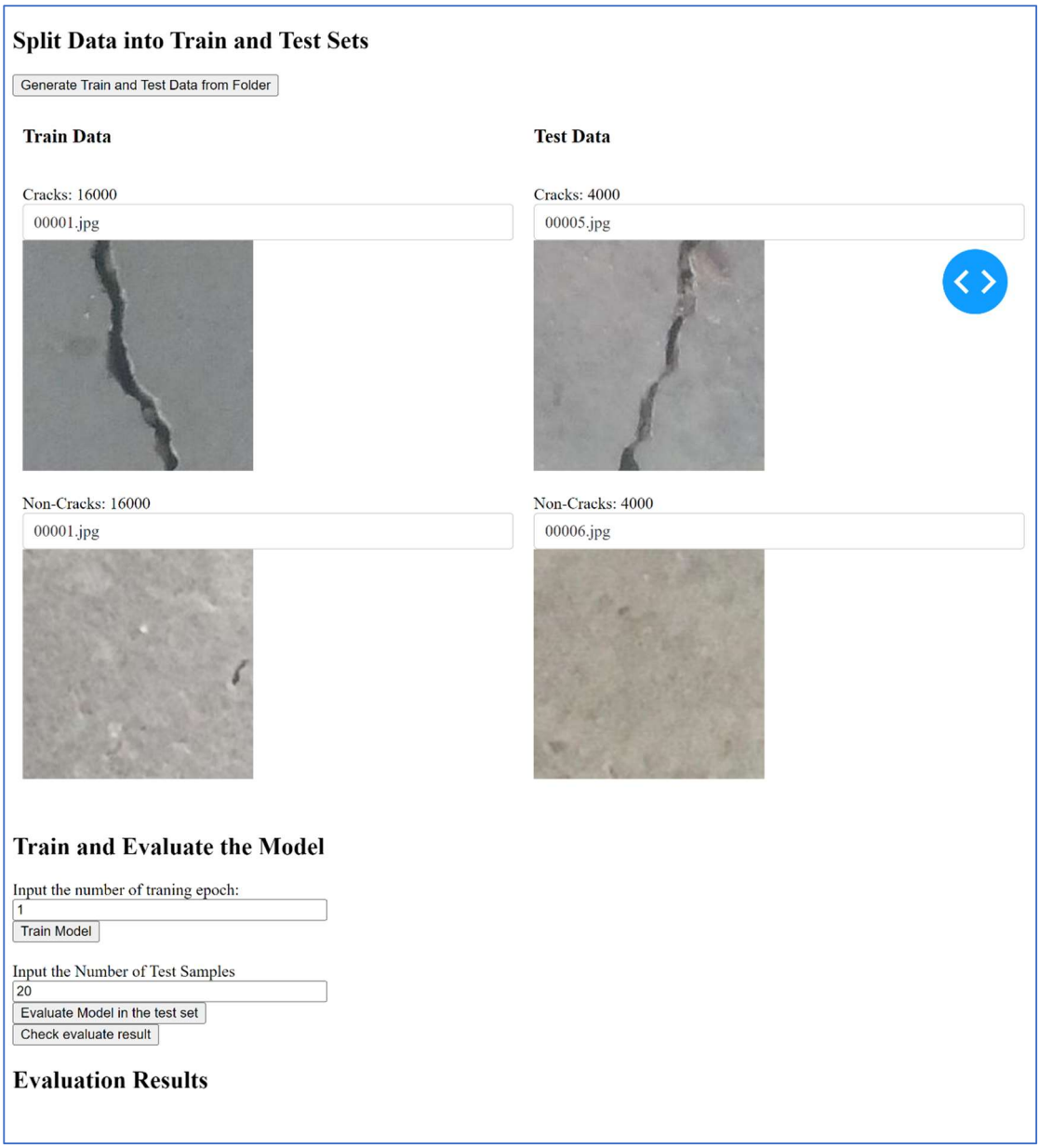


5. Click the Generate Train and Test Data from Folder button. Wait for several minutes. Cracks_splitted8020 will be generated.

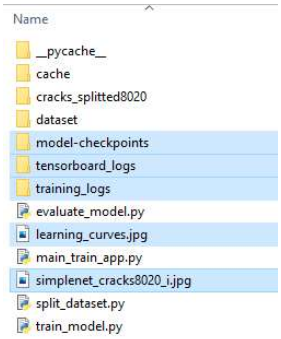




6. Click Check the Train and Test Data.

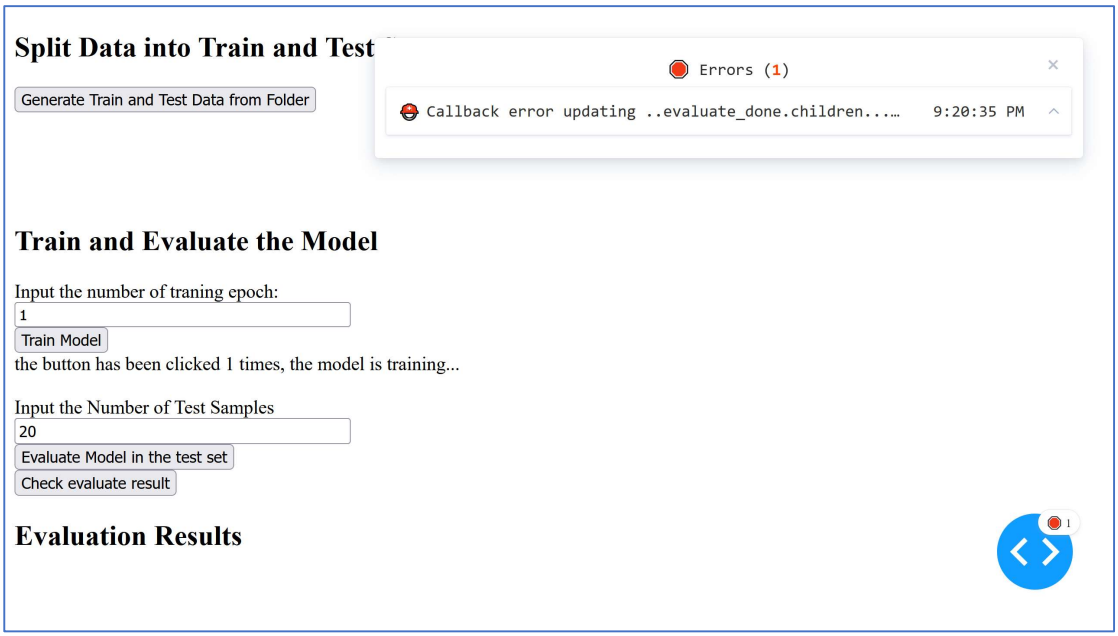


7. Input the number of training epoch, such as 5.
   Estimate the training time: *number of trainings * (1~2) mins*
   To get a well-trained model, generally input >30. Here we just take 5 as an example, which already can train a good model.
   After the model is trained, 5 files will be generated.

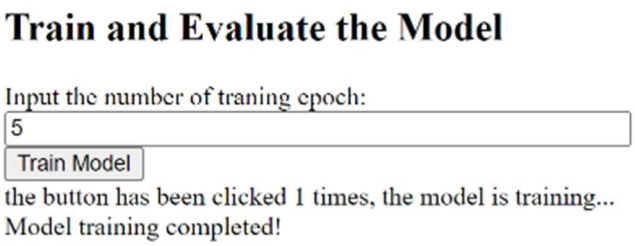


Note:

a) The following errors might arise. But it doesn't influence the workflow. Just ignore it.

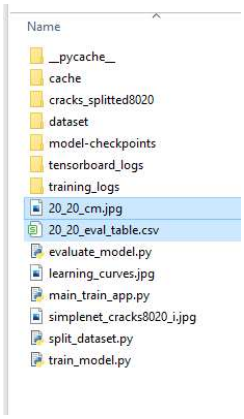


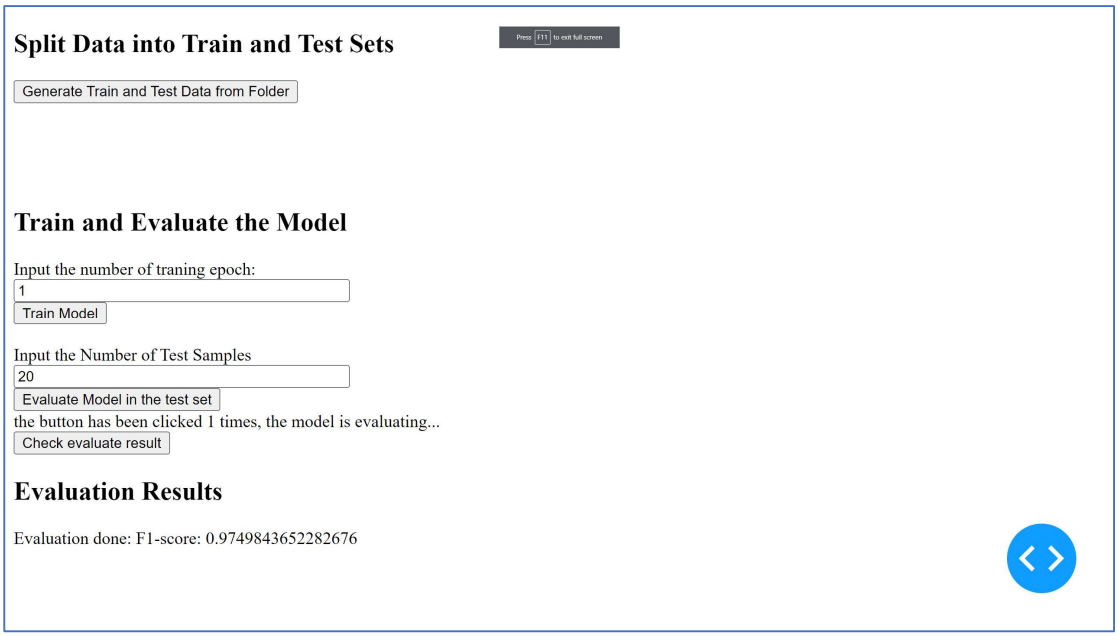
b) After model training is finished, 'Molde training completed!' will show. The webpage might refresh itself. Just ignore it. All files related to the new model has been saved in your work directory. Go to the next step.



c) Model training progress can be seen in your Terminal:

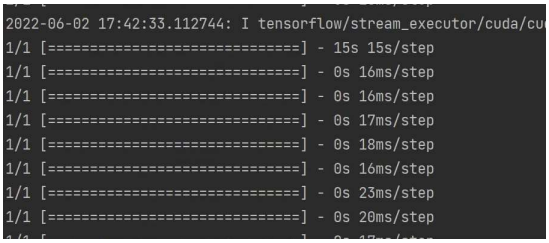
8. Input the number of Test Samples, such as 20 (max: 4000. There will be 4000 cracks images and non-cracks images in the test set). Click Evaluate Model in the test set button. After the evaluation is finished, 2 files will be generated:





Note:

The evaluation progress can also been seen in the Terminal:



9. Click Check evaluation result

   There will be three figures created. The first one is the model structure. The second one is the model loss and accuracy during the training. The last figure shows the confusion matrix of the evaluation. In the table below, we can check the prediction result of each image.

Split Data into Train and Test Sets

Generate Train and Test Data from Folder

Spliting the dataset into train set (80%) and test set (20%)...
Done!

Check the Train and Test Data

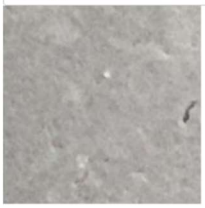
Train Data

Cracks: 16000

00001.jpg

Non-Cracks: 16000

00001.jpg



Test Data

Cracks: 4000

00002.jpg

Non-Cracks: 4000

00004.jpg

Train and Evaluate the Model

Input the number of traning epoch:

5

Train Model

the button has been clicked 1 times, the model is training...
Model training completed!

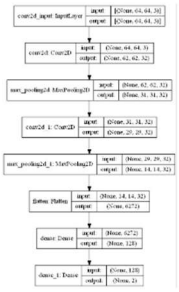
Input the Number of Test Samples

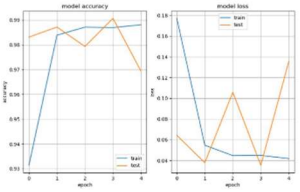
20

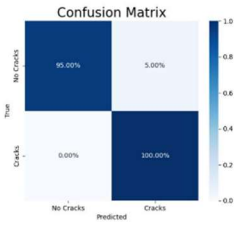
Evaluate Model in the test set

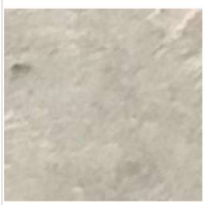
Check evaluate result

Evaluation Results







Inc	filename	Image	True Label	Predicted Label	Correct	Non-Crac	Crack Pr
0	02617.jpg		Non Cracks	Non Cracks	Yes	100	0
1	19113.jpg		Non Cracks	Non Cracks	Yes	100	0

« ‹ 1 / 4 › »