HW5 Yaroslav Popryho

ypopry2@uic.edu

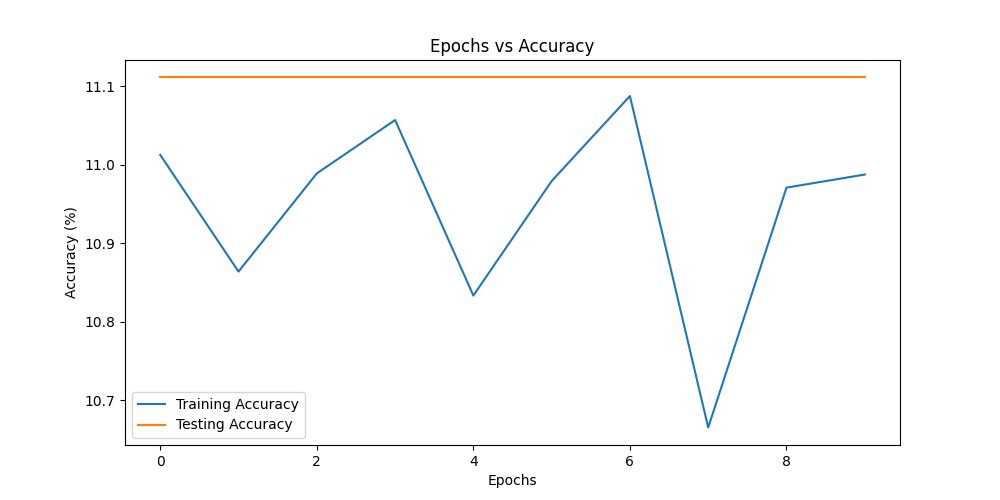
***– The details of your neural network design process, what did you try, what worked, what did not... The optimizer, the loss function, and other hyperparameters that you utilized.***

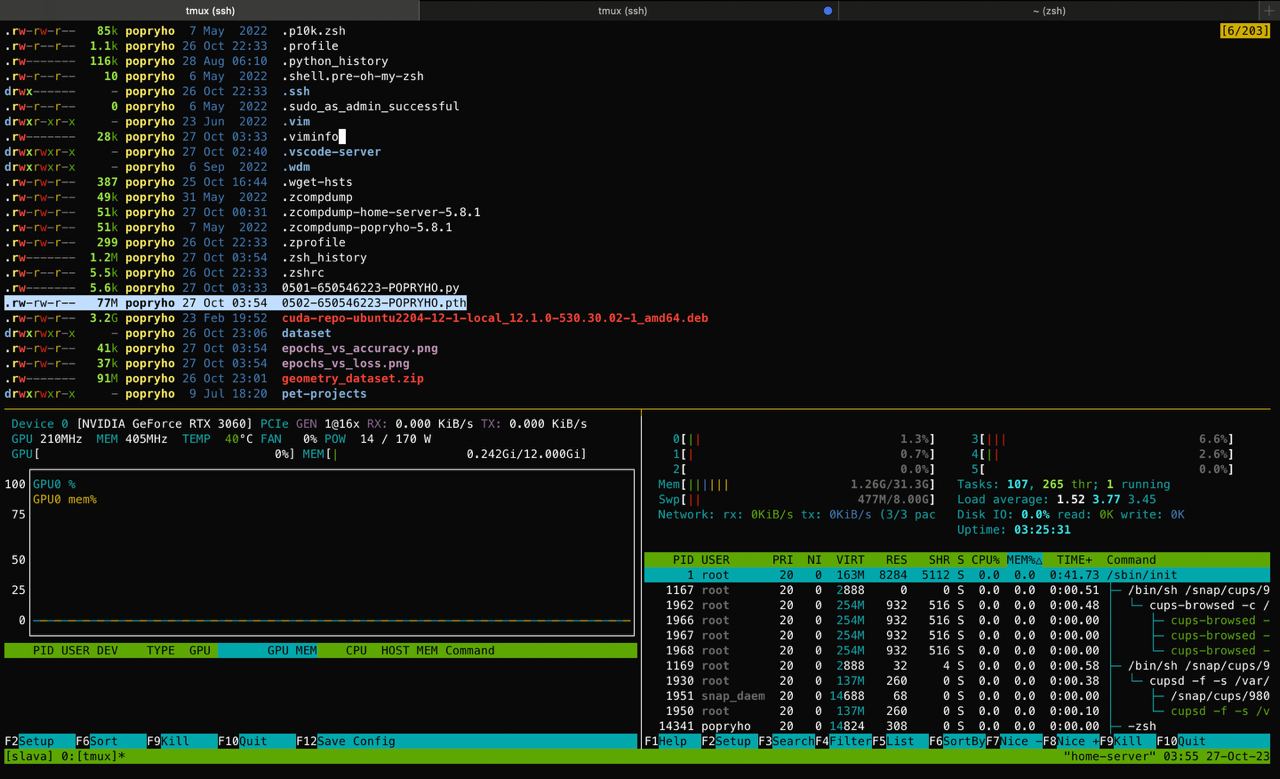
In the first experiment, I utilized the **following** architecture:

* 4 convolutional layers with channels increasing from 32 to 256.
* 3 max-pooling layers.
* 2 fully connected layers, with a dropout of 50% before the final layer.

Hyper parameters:

* Learning rate: 0.001.
* Batch size: 132.
* Optimizer: Adam
* Loss function: Cross Entropy Loss



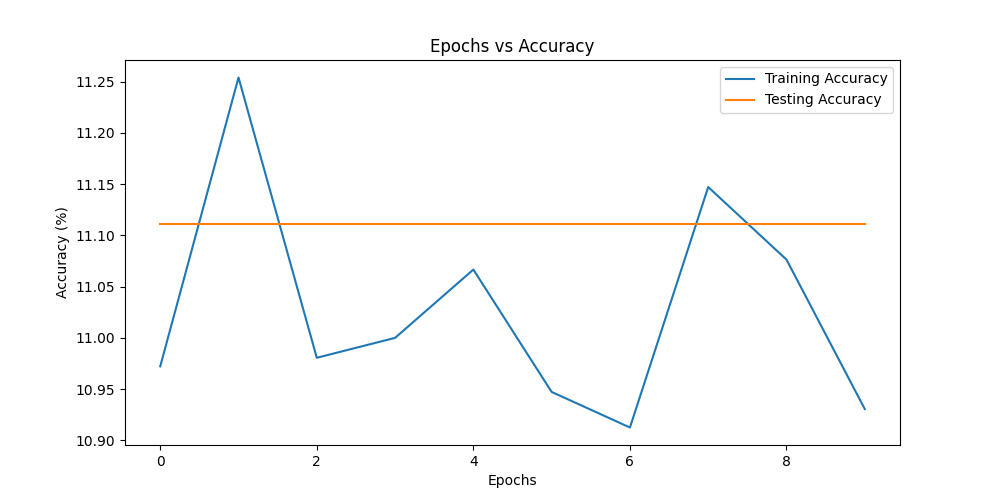


**Result**: The loss on the test set was higher than on the training set, indicating overfitting. Surprisingly, the training accuracy was also lower than expected. Additionally, the model size (77 MB) exceeded the acceptable limit.

For the 2nd experiment, I was using the same **model** architecture from experiment 1, but just modified the training parameters:

* Set learning rate to 0.01.

**Result**: The testing accuracy remained constant across epochs, suggesting that the model wasn't learning effectively or had reached a local minimum.



I then opted for a simpler model with following changes:

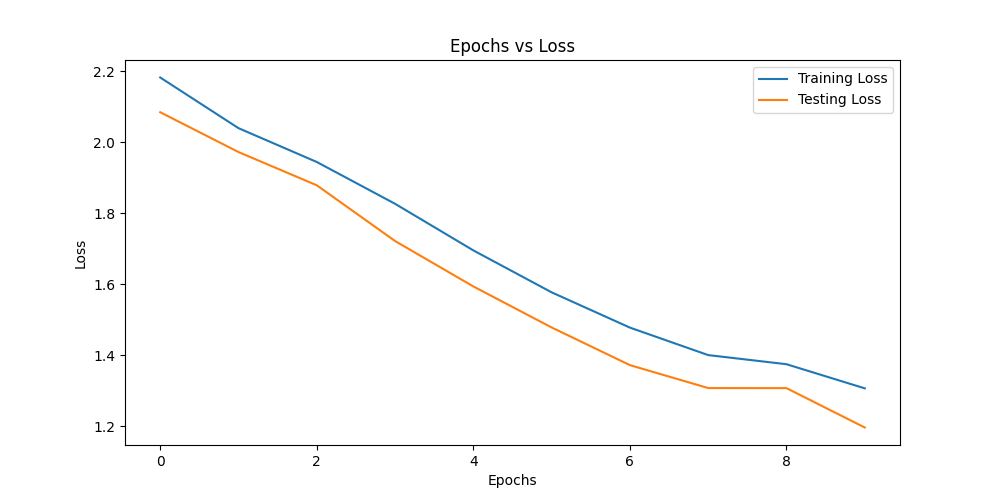
* Reduced the number of channels in convolutional layers.
* Removed one convolutional layer.
* Adjusted the size of the fully connected layers.

Hyper parameters:

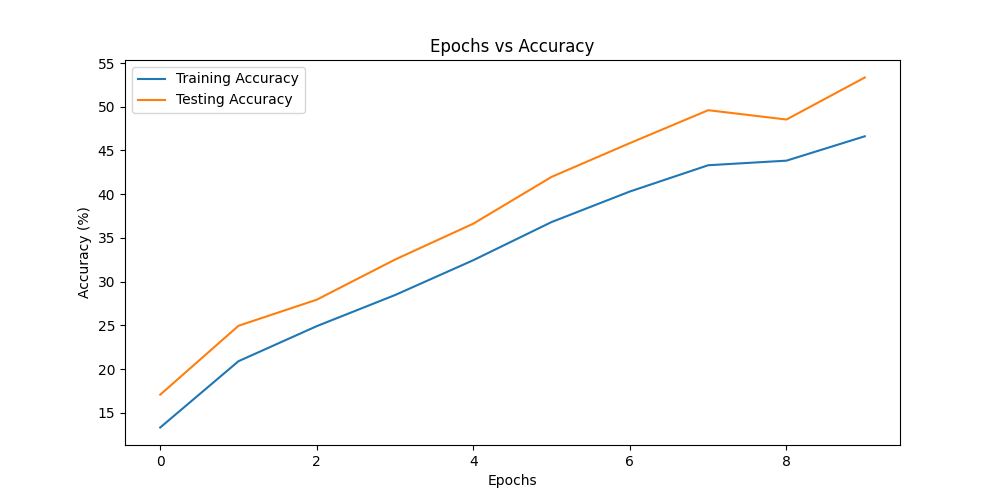
* Batch size: 32.
* Learning rate: 0.001.

**Result**: This model showed the expected behavior with increasing accuracy and decreasing loss across epochs for both training and testing sets, signifying improved generalization and performance.

***– A graph that shows epochs vs loss on training set, and on the same graph epoch vs loss on the set set.***

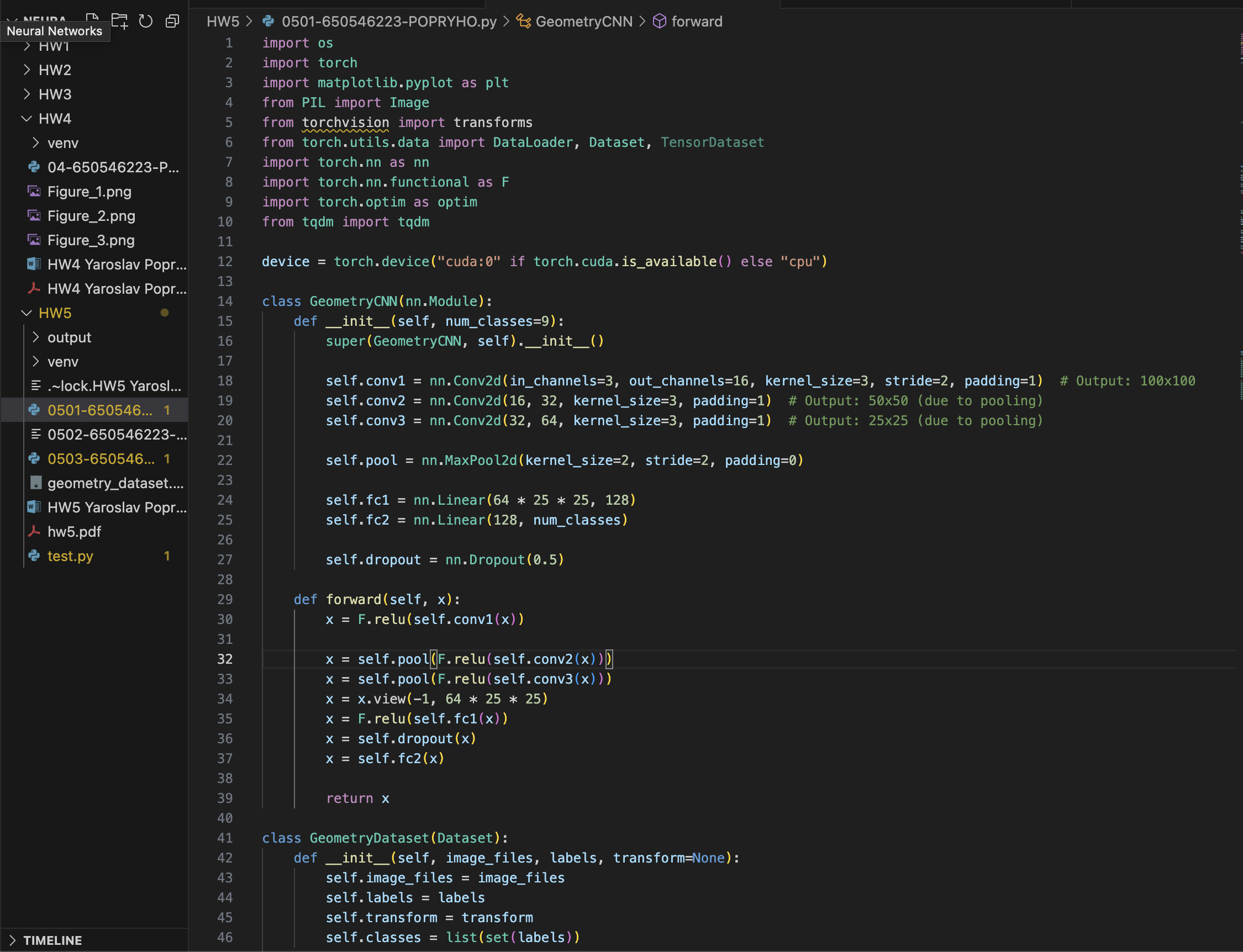


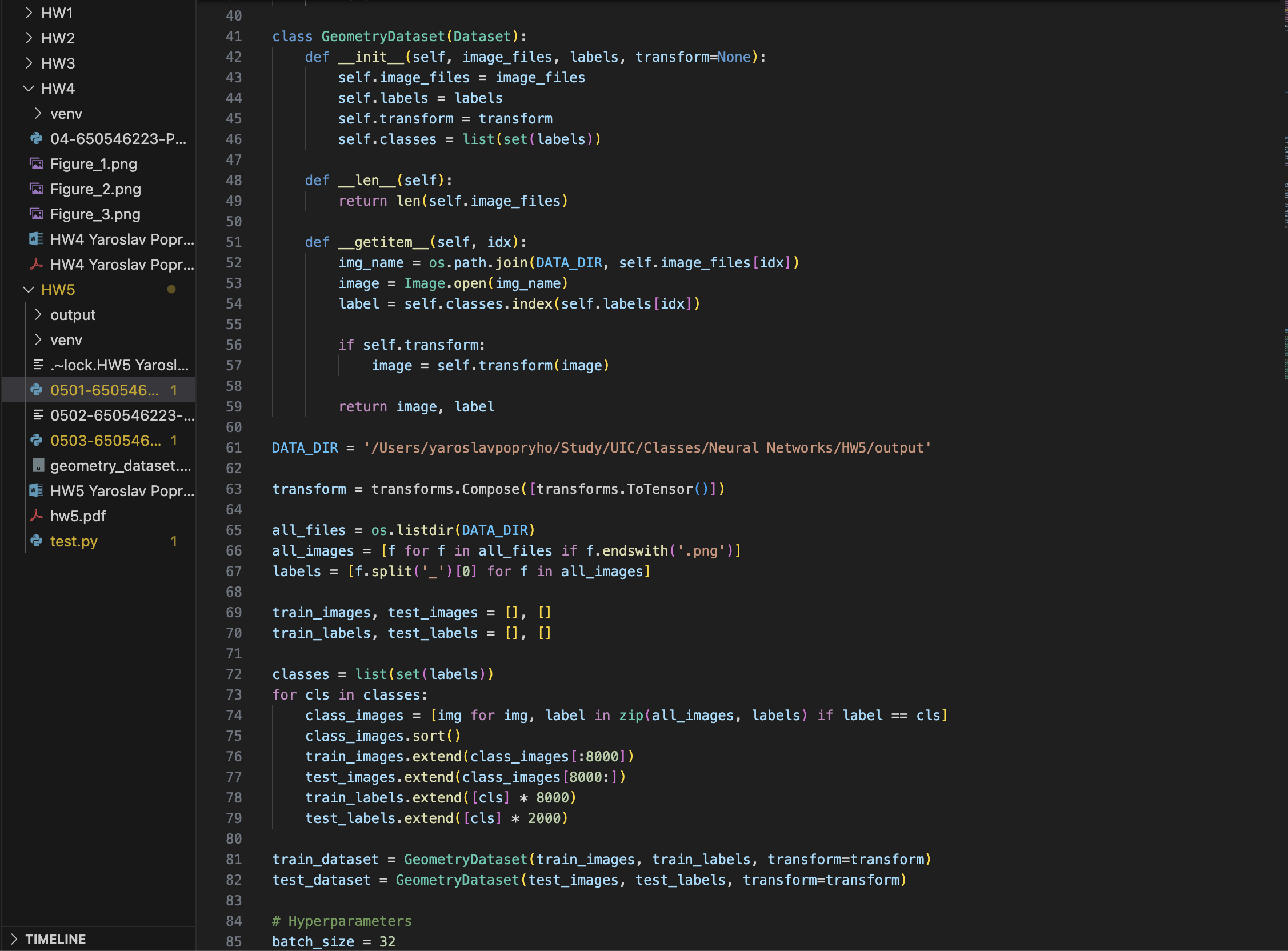
***– A graph that shows epochs vs accuracy on training set, and on the same graph epoch vs accuracy on the set set. – All your codes***

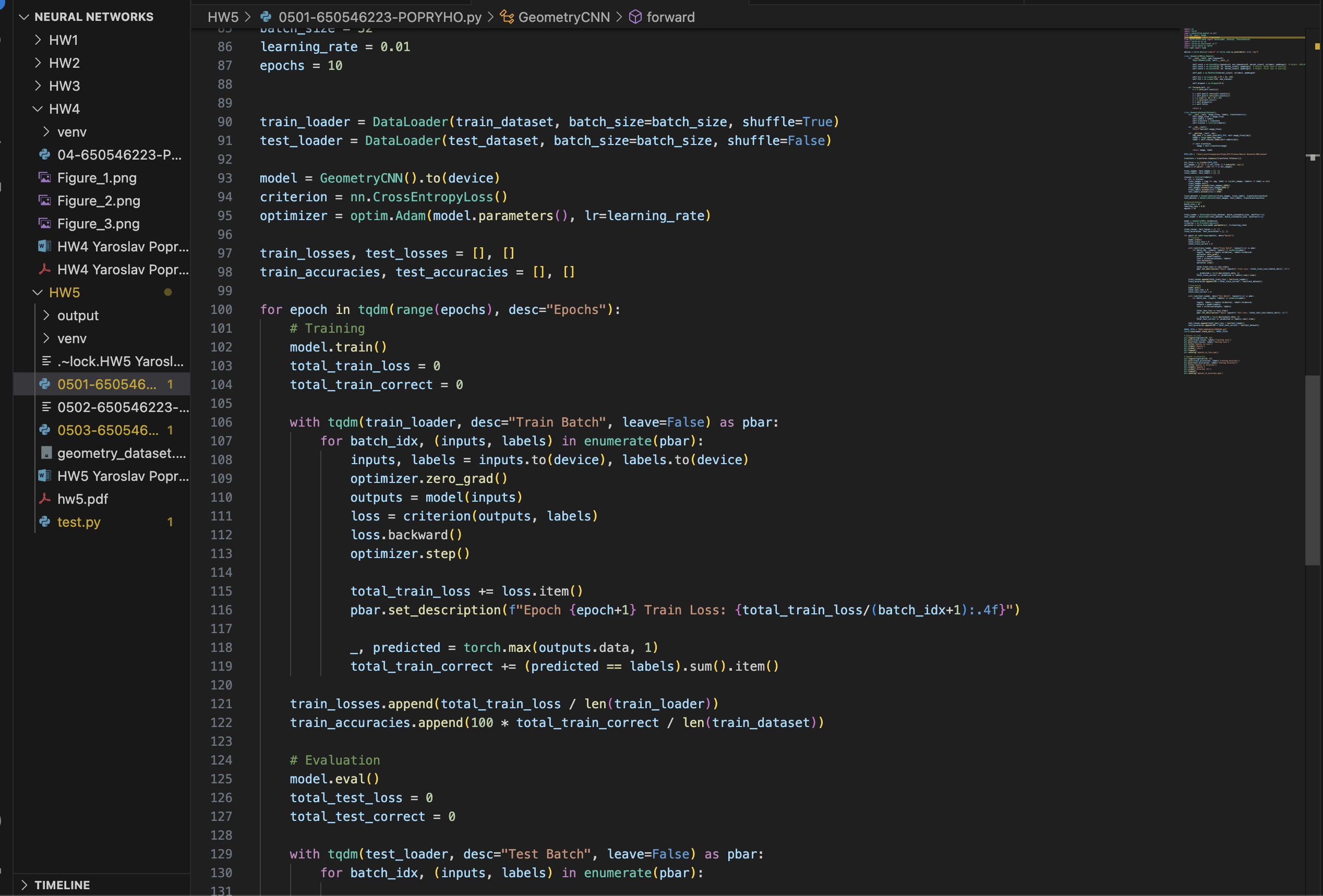


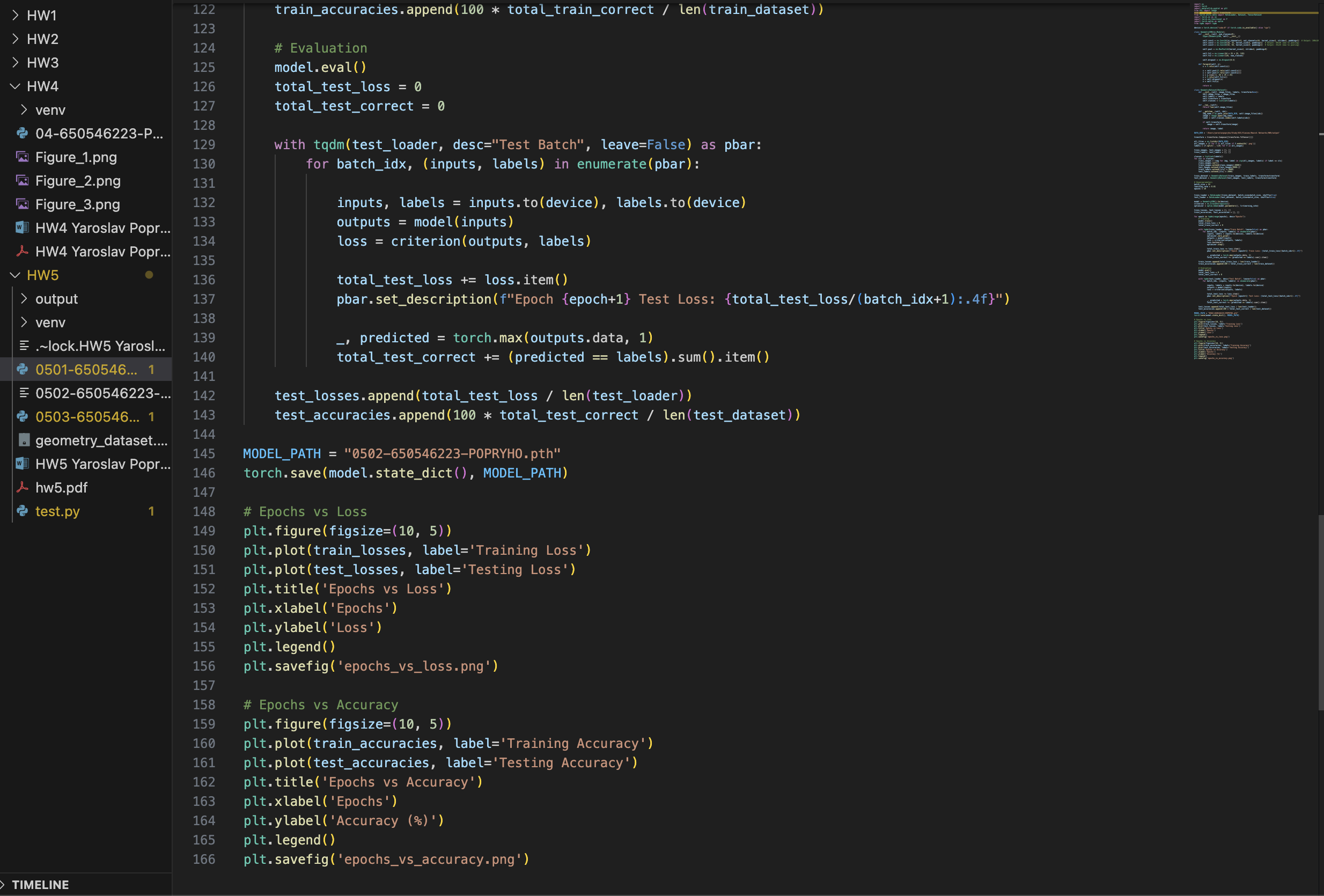
***– All your codes.***

****0501-650546223-POPRYHO.py:****

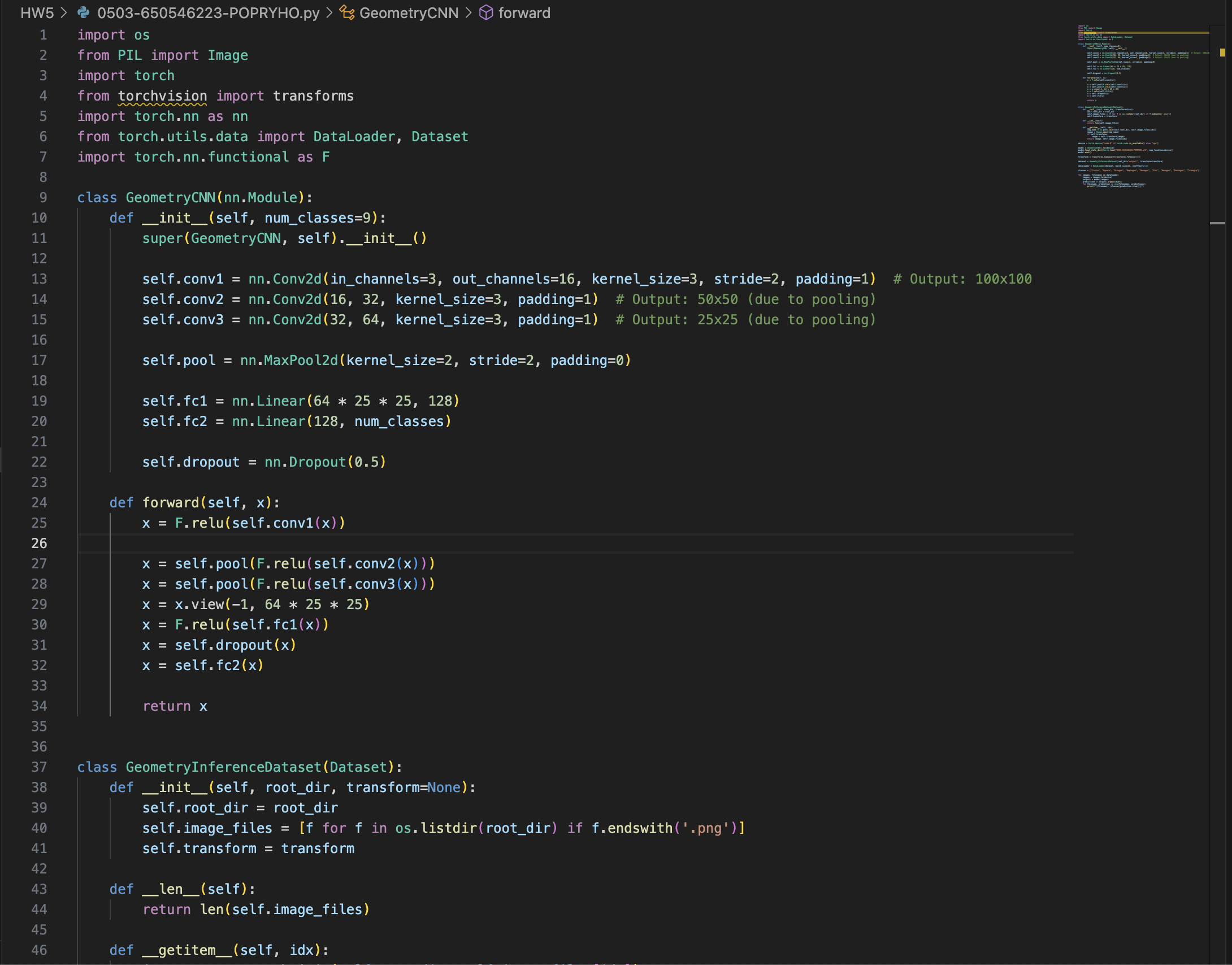


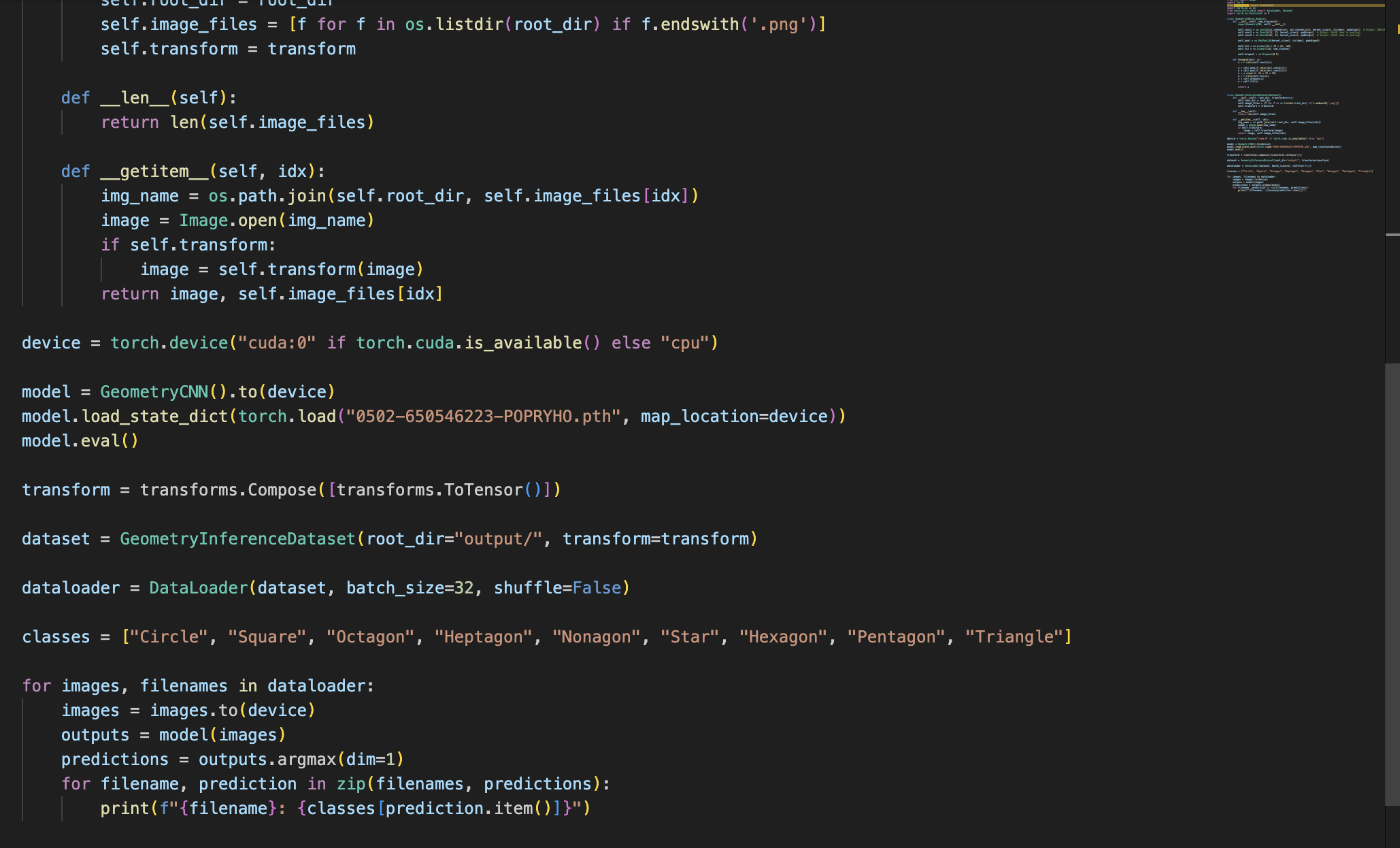






**0503-650546223-POPRYHO.py**

****

****