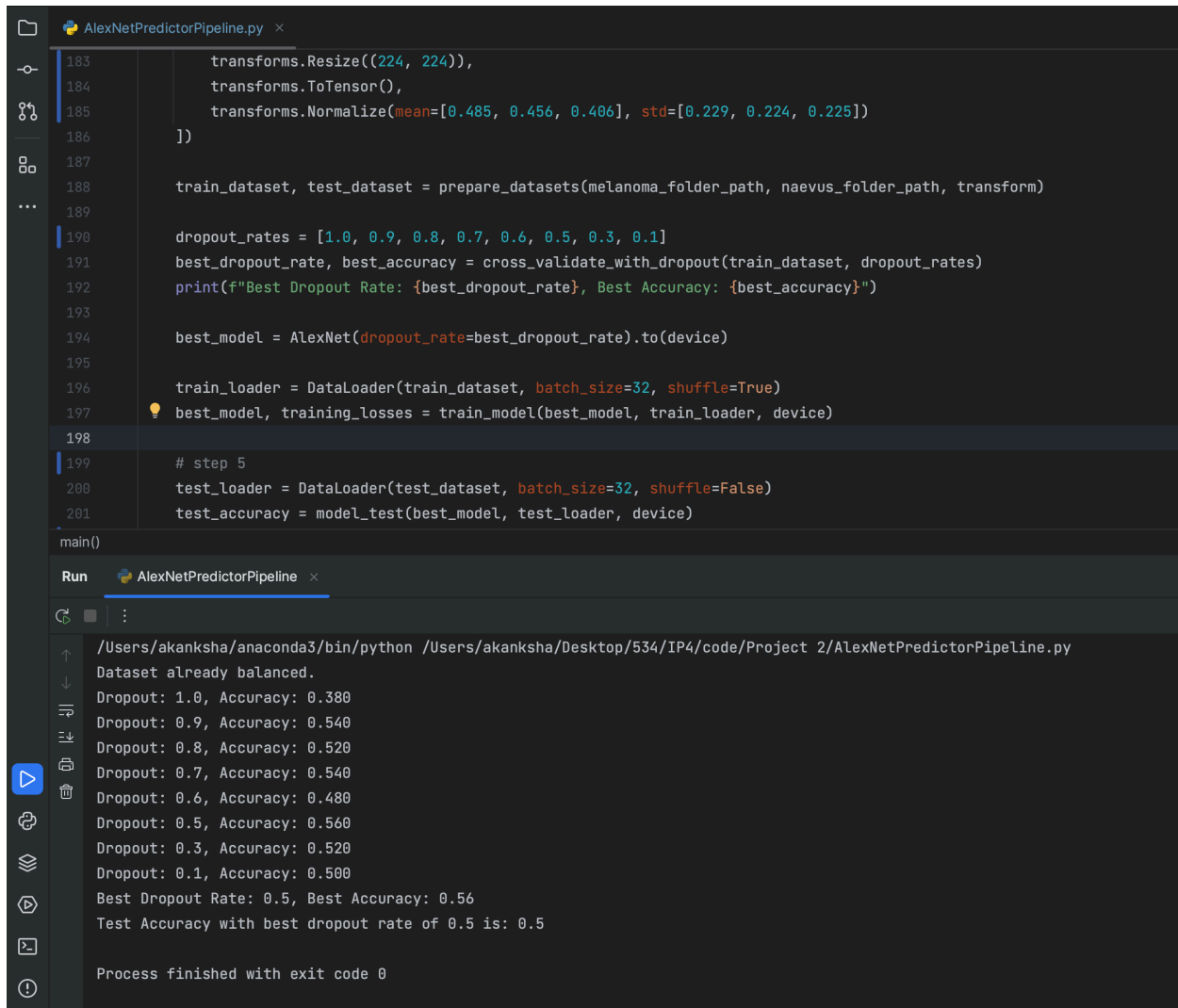


I am splitting training and testing data at runtime, so I have not stored them in different (train and test) folders.

The data folder is already balanced but my code includes the script that I used to balance it.

The code is pretty self-explanatory hence there aren't many comments.



```
183     transforms.Resize((224, 224)),
184     transforms.ToTensor(),
185     transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
186 ])
187
188 train_dataset, test_dataset = prepare_datasets(melanoma_folder_path, naevus_folder_path, transform)
189
190 dropout_rates = [1.0, 0.9, 0.8, 0.7, 0.6, 0.5, 0.3, 0.1]
191 best_dropout_rate, best_accuracy = cross_validate_with_dropout(train_dataset, dropout_rates)
192 print(f"Best Dropout Rate: {best_dropout_rate}, Best Accuracy: {best_accuracy}")
193
194 best_model = AlexNet(dropout_rate=best_dropout_rate).to(device)
195
196 train_loader = DataLoader(train_dataset, batch_size=32, shuffle=True)
197 best_model, training_losses = train_model(best_model, train_loader, device)
198
199 # step 5
200 test_loader = DataLoader(test_dataset, batch_size=32, shuffle=False)
201 test_accuracy = model_test(best_model, test_loader, device)
202
203 main()
```

Run AlexNetPredictorPipeline

```
/Users/akanksha/anaconda3/bin/python /Users/akanksha/Desktop/534/IP4/code/Project 2/AlexNetPredictorPipeline.py
Dataset already balanced.
Dropout: 1.0, Accuracy: 0.380
Dropout: 0.9, Accuracy: 0.540
Dropout: 0.8, Accuracy: 0.520
Dropout: 0.7, Accuracy: 0.540
Dropout: 0.6, Accuracy: 0.480
Dropout: 0.5, Accuracy: 0.560
Dropout: 0.3, Accuracy: 0.520
Dropout: 0.1, Accuracy: 0.500
Best Dropout Rate: 0.5, Best Accuracy: 0.56
Test Accuracy with best dropout rate of 0.5 is: 0.5

Process finished with exit code 0
```