STAT 390 Final Presentation

Akhil, David, Harvey, Jeffrey, **Veer**

Testing Methodology

Model Testing Framework

1) Begin with base pre-trained network
2) Train models on a per-stain basis
3) Tune learning rate + number epochs until consistent convergence is reached
4) Use model evaluation + model architecture to identify hyperparameters to manipulate (with a focus on batch_size)
5) Focused on evaluating patch-level metrics

Note on RAM Constraints

- Initial plan – ResNet50 with CBAM attention modules
- RAM overload even with single epoch training, persisting at low batch sizes of 8/16
- Attempt at batch size = 4 and without CBAM solved problem momentarily
- Small batch size caused problems of noisy gradients, with major spikes in training and validation metric curves
- Current setup makes it difficult to train more complex networks with a big data set at large batch sizes, especially if we want to unfreeze more layers

ResNet50

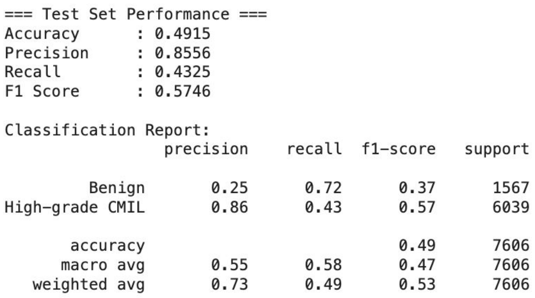
ResNet Overview

- ResNet is considered a strong baseline for histopathology image classification
- Its residual connections allow training of very deep networks, helping to capture complex tissue features. Its residual blocks include skip connections to help address vanishing gradient problems
- ResNet models are pre-trained on ImageNet data (ResNet50)
- ResNet50 has 50 layers, organized into 4 stages of convolutional blocks
- Has an in-built Adaptive Pooling layer
- Parameters ~23.5 million

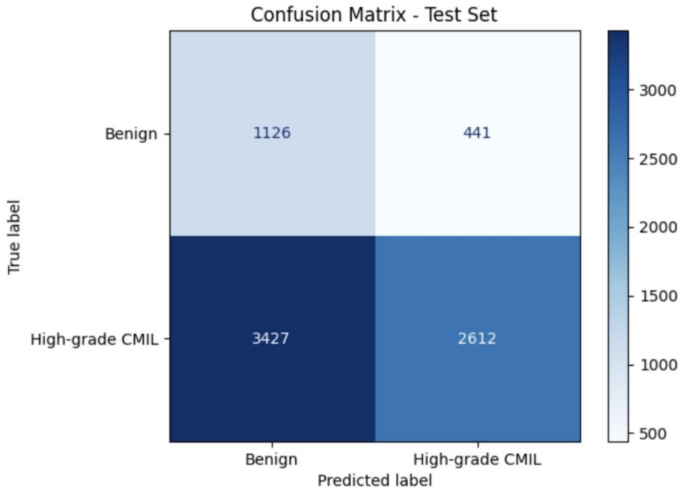
Variant 1 Specifications

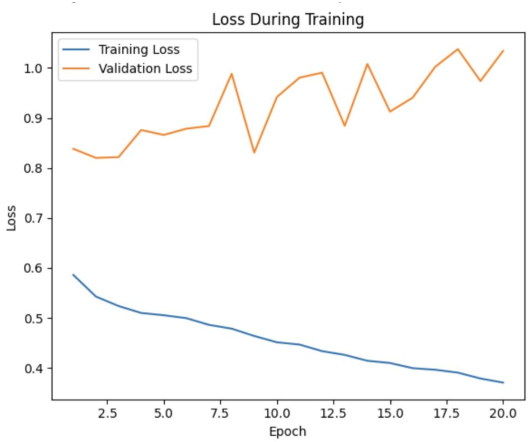
- Trained base ResNet50 for each stain using in-built Adaptive Pooling layer
- Removed Max Pooling layers to prevent image size compatibility issues for the smaller patches
- Batch size = 4
- Number of epochs = 20
- Learning rate = 1.00E-04
- Overall, model performed poorly (except for sox10), failing to identify features associated with high grade C-MIL

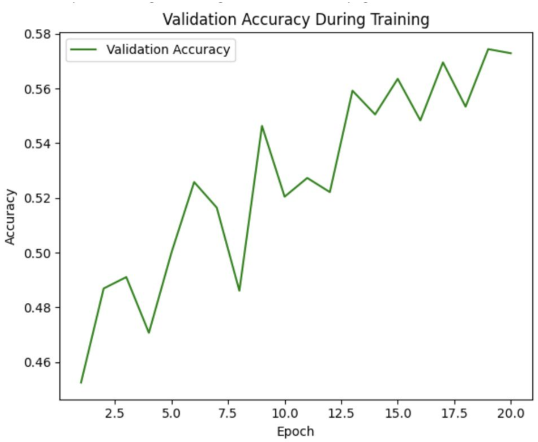
Variant 1: H&E



```
=== Test Set Performance ===
Accuracy      : 0.4915
Precision     : 0.8556
Recall        : 0.4325
F1 Score      : 0.5746

Classification Report:
                precision    recall  f1-score   support

        Benign       0.25      0.72      0.37      1567
High-grade CMIL       0.86      0.43      0.57      6039

      accuracy                           0.49      7606
     macro avg       0.55      0.58      0.47      7606
  weighted avg       0.73      0.49      0.53      7606
```

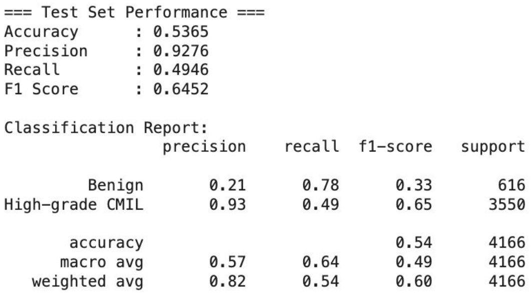




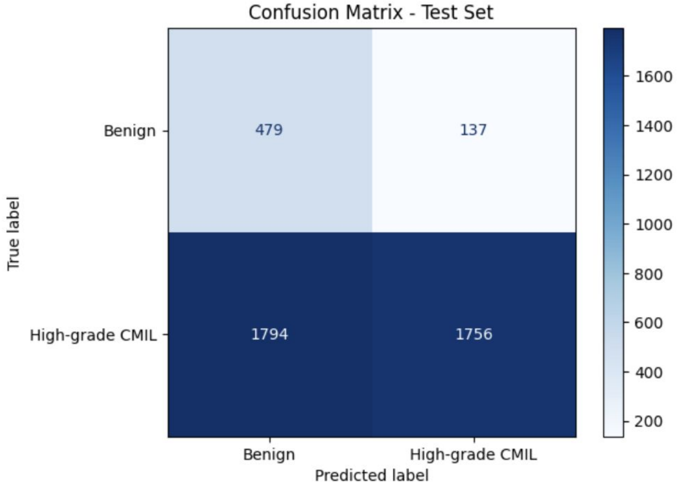


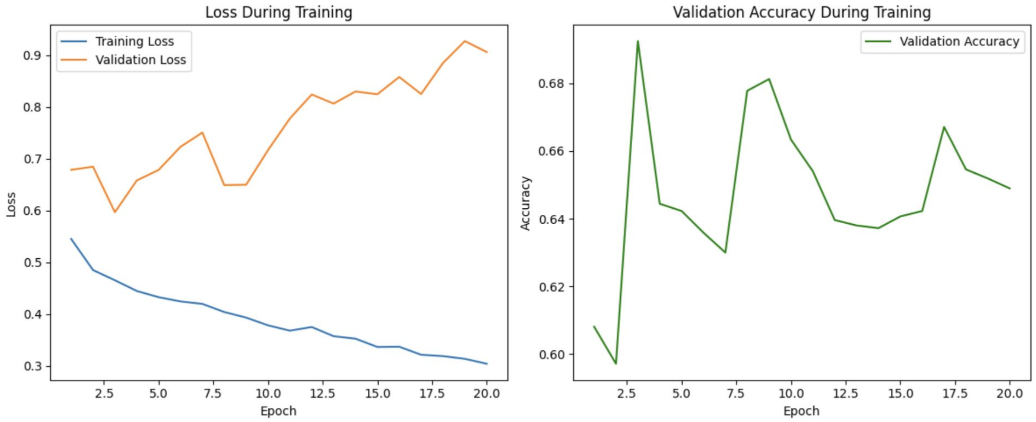
Low batch size leads to very noisy gradients, as shown by volatile nature of validation accuracy curve

Variant 1: Melan



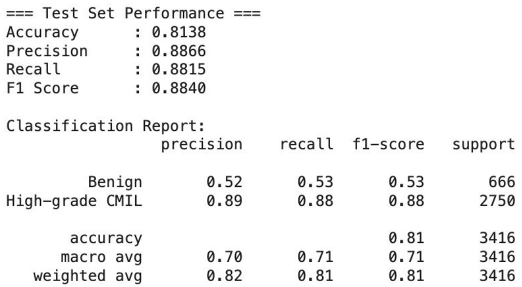
```
=== Test Set Performance ===
Accuracy      : 0.5365
Precision     : 0.9276
Recall        : 0.4946
F1 Score      : 0.6452

Classification Report:
                precision    recall  f1-score   support

         Benign       0.21      0.78      0.33       616
High-grade CMIL       0.93      0.49      0.65      3550

       accuracy                           0.54      4166
      macro avg       0.57      0.64      0.49      4166
   weighted avg       0.82      0.54      0.60      4166
```



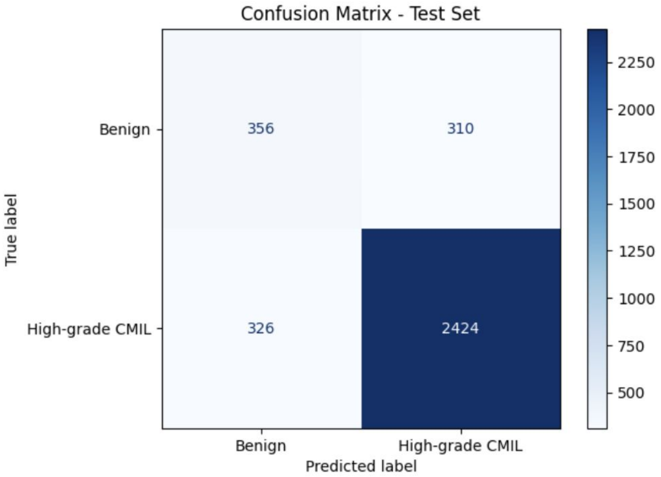


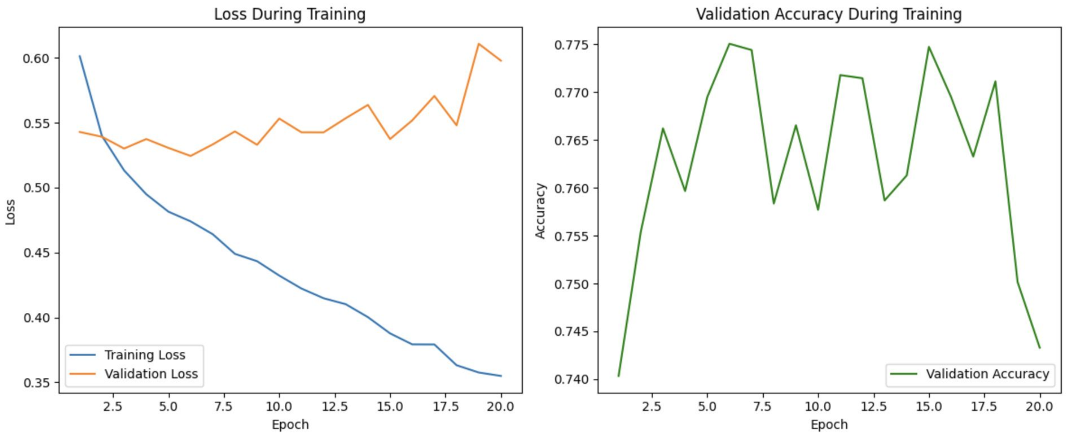
Again, really noisy gradients and high instability in validation performance. Model not generalizing well at all and probably failing to converge

Variant 1: Sox10



```
=== Test Set Performance ===
Accuracy     : 0.8138
Precision    : 0.8866
Recall       : 0.8815
F1 Score     : 0.8840

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.52      0.53      0.53       666
High-grade CMIL       0.89      0.88      0.88      2750

       accuracy                           0.81      3416
      macro avg       0.70      0.71      0.71      3416
   weighted avg       0.82      0.81      0.81      3416
```





Surprisingly high-grade recall given poor performance for h&e and melan stains. Validation performance seems less unstable for sox10 compared to other 2

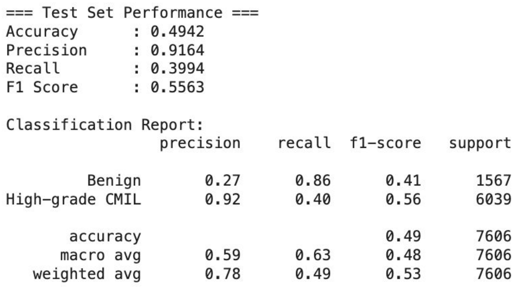
Variant 1 Comments

- H&E and Melan performance clearly not up to the mark, with recall even below a 50/50 split
- Model seems to be better at identifying characteristics of high grade C-MIL in the sox10 stain, with a high recall of 0.88
- However, validation performance still seemed to be noisy, and convergence not seen clearly
- Can explore variants of this model for sox10 stain specifically, and retry on h&e and melan if see good results

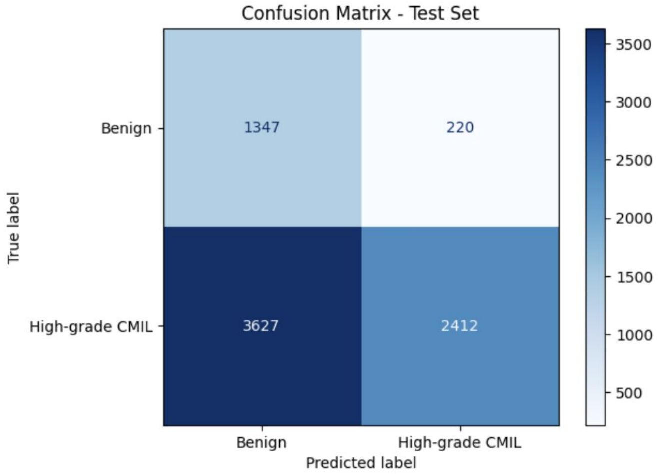
Variant 2 Specifications

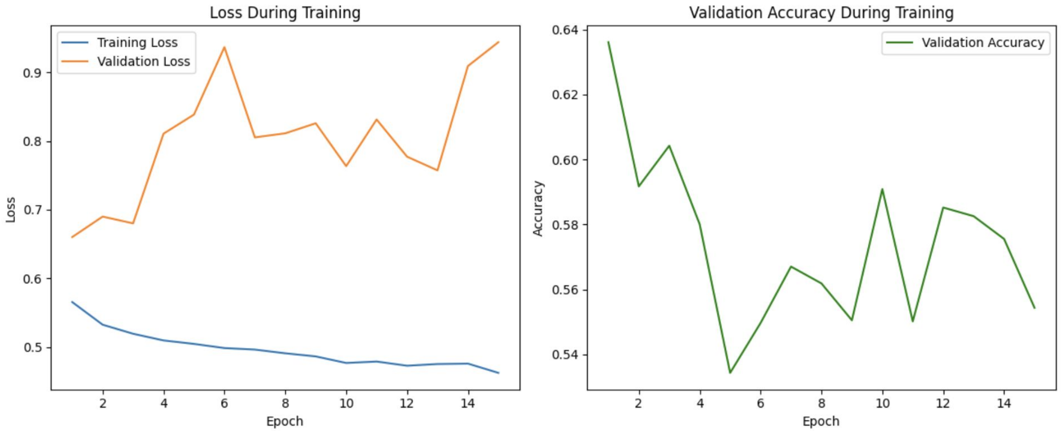
- Resized all images to 224x224, while still keeping base Adaptive Pooling layer in ResNet50 architecture
- Reintroduced Max Pooling layers
- Batch size = 4
- Number of epochs = 15
- Learning rate = 1.00E-04
- See more promising results across all 3 stains, suggesting that resizing may be the better choice going forward

Variant 2: H&E



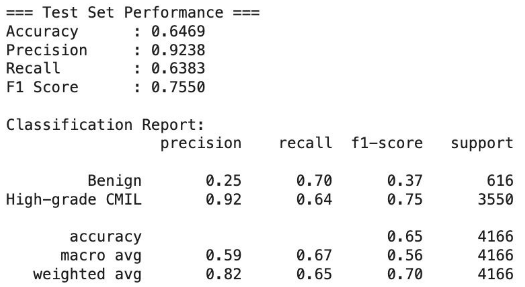
```
=== Test Set Performance ===
Accuracy      : 0.4942
Precision     : 0.9164
Recall        : 0.3994
F1 Score      : 0.5563

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.27      0.86      0.41      1567
High-grade CMIL       0.92      0.40      0.56      6039

       accuracy                           0.49      7606
      macro avg       0.59      0.63      0.48      7606
   weighted avg       0.78      0.49      0.53      7606
```



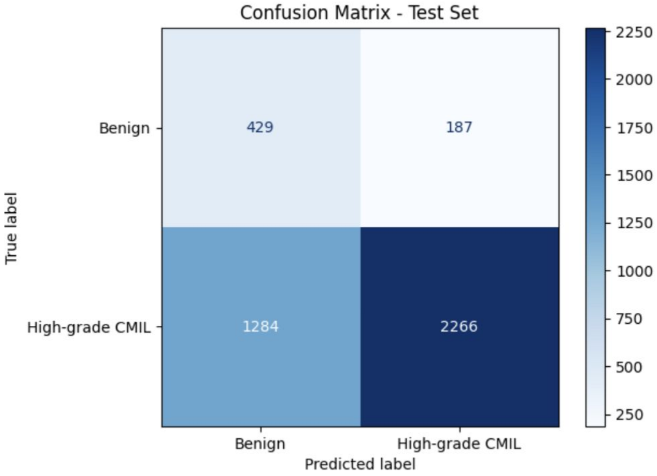


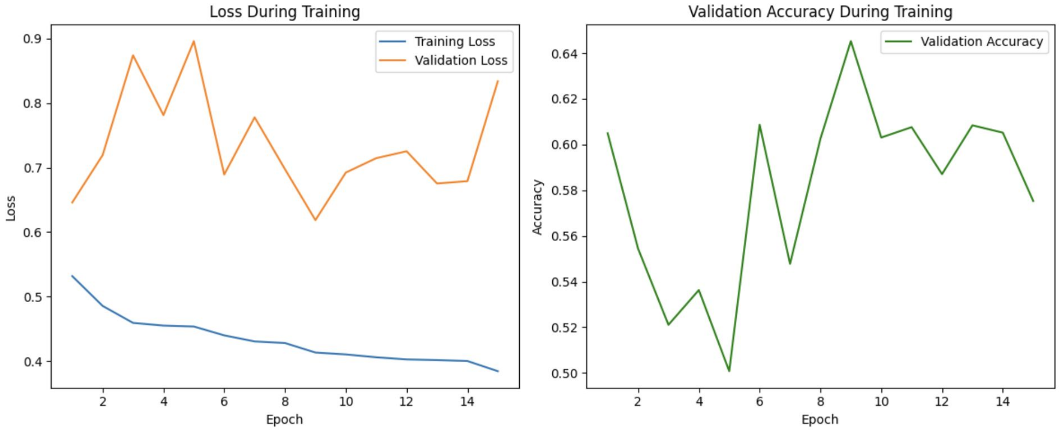
H&E model performed poorly again, suggesting a mix of low batch size + nature of h&e stain causing problems

Variant 2: Melan



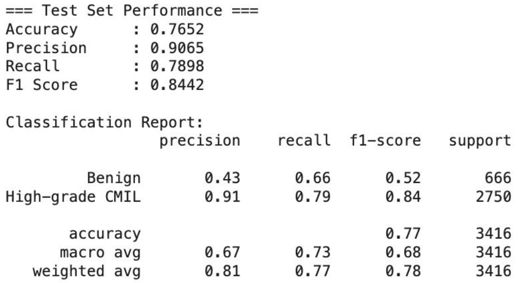
```
=== Test Set Performance ===
Accuracy      : 0.6469
Precision     : 0.9238
Recall        : 0.6383
F1 Score      : 0.7550

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.25      0.70      0.37       616
High-grade CMIL       0.92      0.64      0.75      3550

       accuracy                           0.65      4166
      macro avg       0.59      0.67      0.56      4166
   weighted avg       0.82      0.65      0.70      4166
```



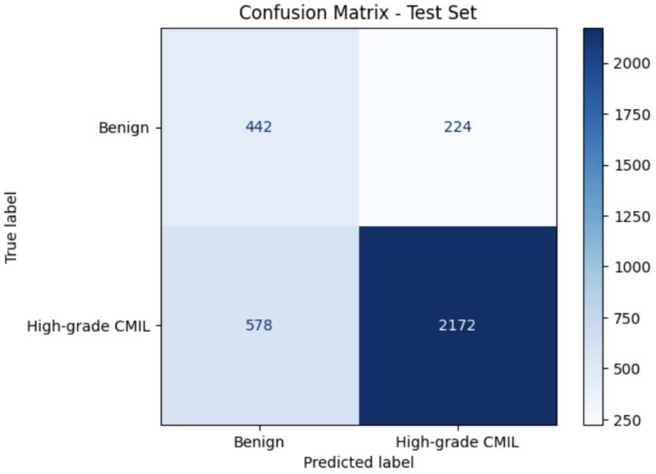


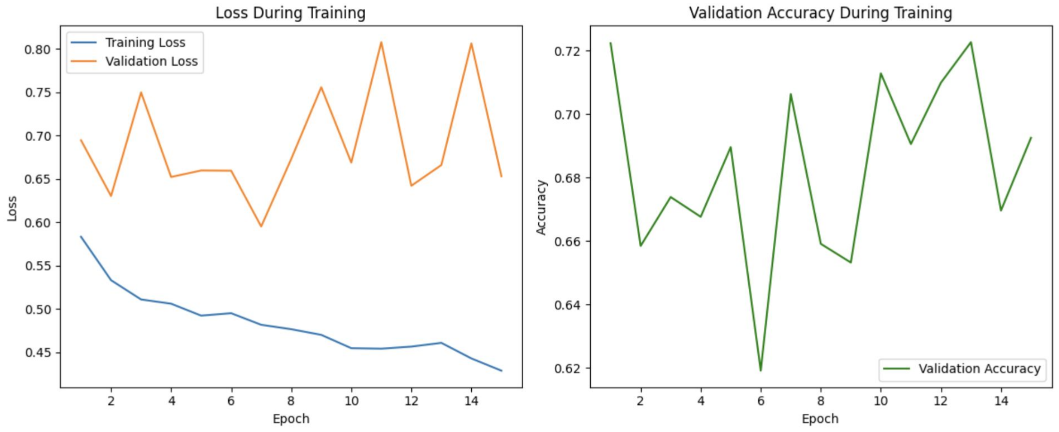
Improved performance relative to baseline adaptive pooling model on melan stain. However, low batch size continues to deliver instability while training

Variant 2: Sox10



```
=== Test Set Performance ===
Accuracy      : 0.7652
Precision     : 0.9065
Recall        : 0.7898
F1 Score      : 0.8442

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.43      0.66      0.52       666
High-grade CMIL       0.91      0.79      0.84      2750

       accuracy                           0.77      3416
      macro avg       0.67      0.73      0.68      3416
   weighted avg       0.81      0.77      0.78      3416
```





Lower sox10 accuracy compared to previous model, but still best overall out of all 3 stains

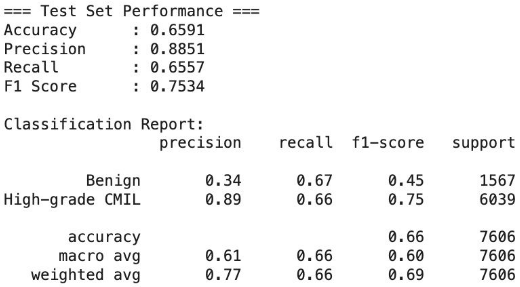
Variant 2 Comments

- Sox10-trained model continues to be the best performer
- Resizing seems to result in slightly better test performance, as seen by the improvement in melan stain
- A lot of the problems could be attributed to low batch size and the resultant noisy gradients
- Being able to unfreeze more layers should further help the model identify the more nuanced features defining C-MIL classes

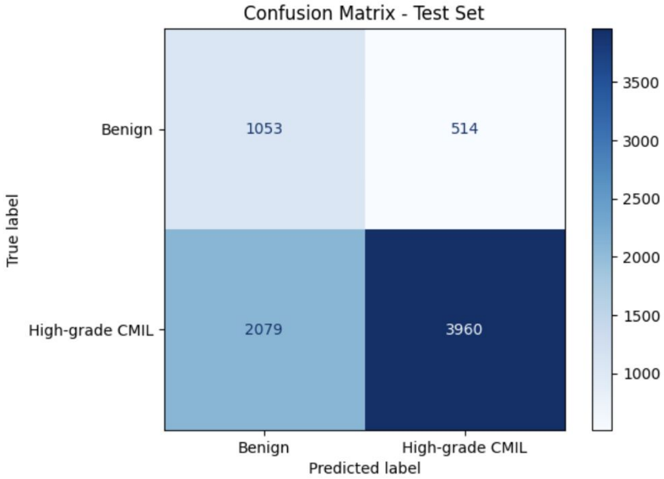
Variant 3 Specifications

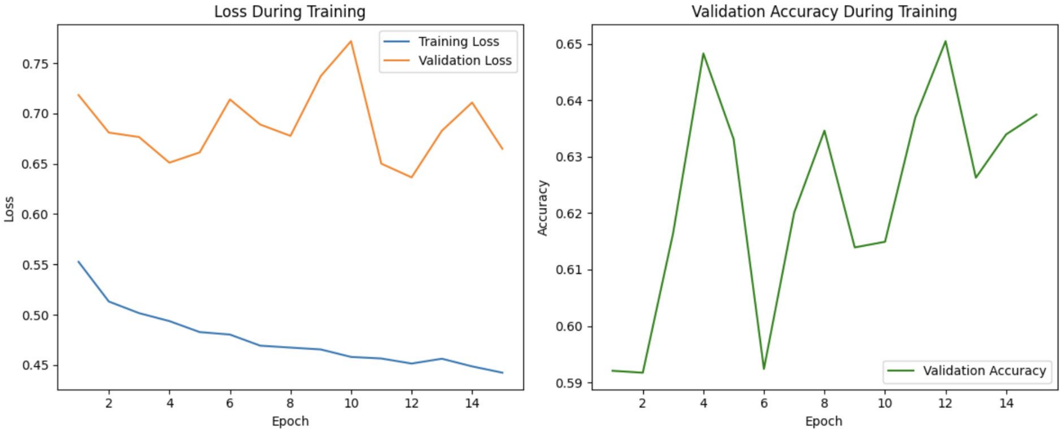
- Similar approach to variant 2, resizing all images to 224x224
- Kept Max Pooling layers
- Increased batch size to 8
- Number of epochs = 15
- Introduced data augmentation
- Increased learning rate to 5.00E-04 as a complement to higher batch size

Variant 3: H&E



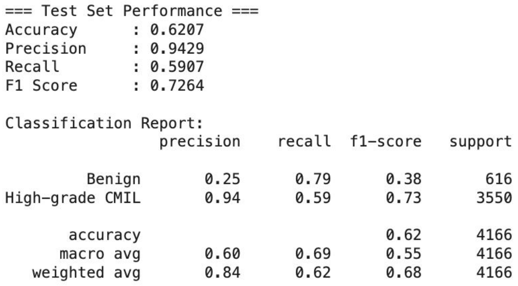
```
=== Test Set Performance ===
Accuracy      : 0.6591
Precision     : 0.8851
Recall        : 0.6557
F1 Score      : 0.7534

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.34      0.67      0.45      1567
High-grade CMIL       0.89      0.66      0.75      6039

       accuracy                           0.66      7606
      macro avg       0.61      0.66      0.60      7606
   weighted avg       0.77      0.66      0.69      7606
```



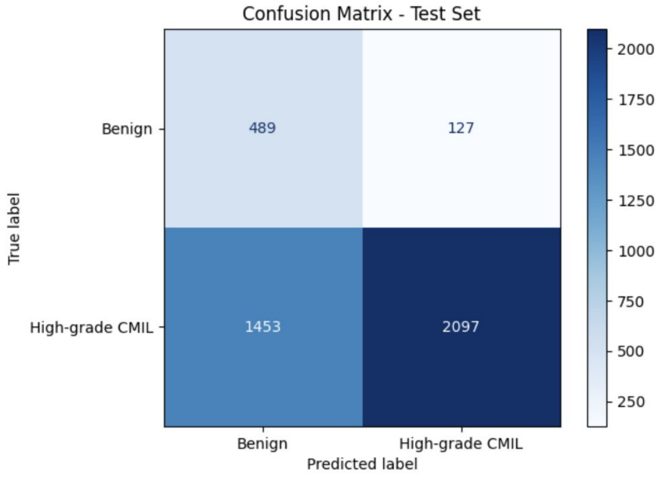


Huge jump in h&e performance, indicating batch size plays a crucial role and is an important parameter to tune

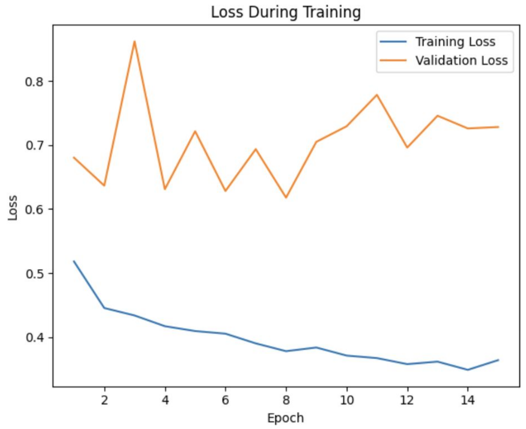
Variant 3: Melan



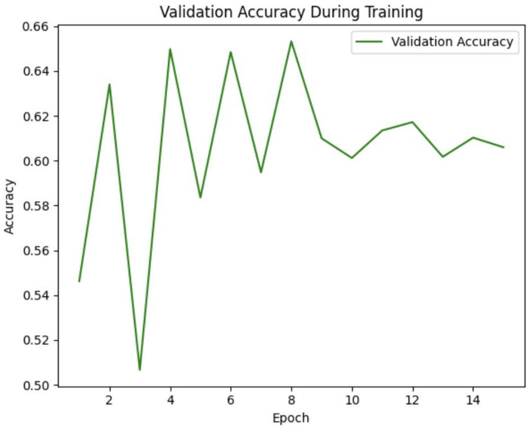
```
=== Test Set Performance ===
Accuracy      : 0.6207
Precision     : 0.9429
Recall        : 0.5907
F1 Score      : 0.7264

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.25      0.79      0.38       616
High-grade CMIL       0.94      0.59      0.73      3550

       accuracy                           0.62      4166
      macro avg       0.60      0.69      0.55      4166
   weighted avg       0.84      0.62      0.68      4166
```



Confusion Matrix - Test Set



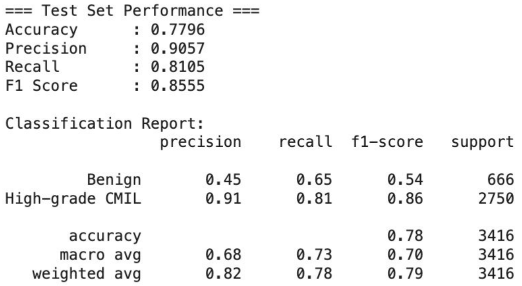
Loss During Training



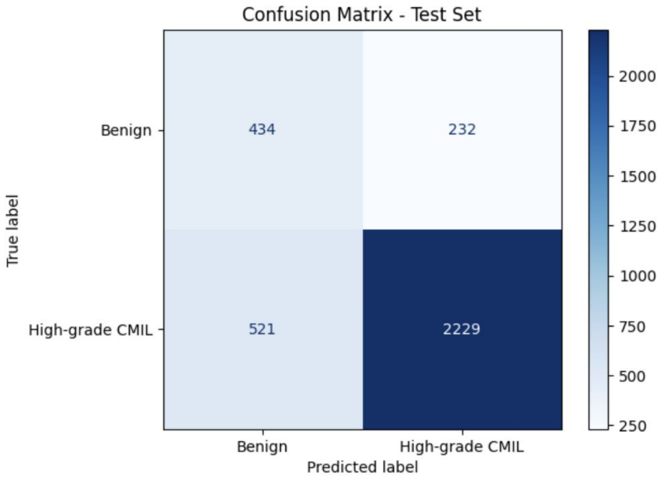
Validation Accuracy During Training

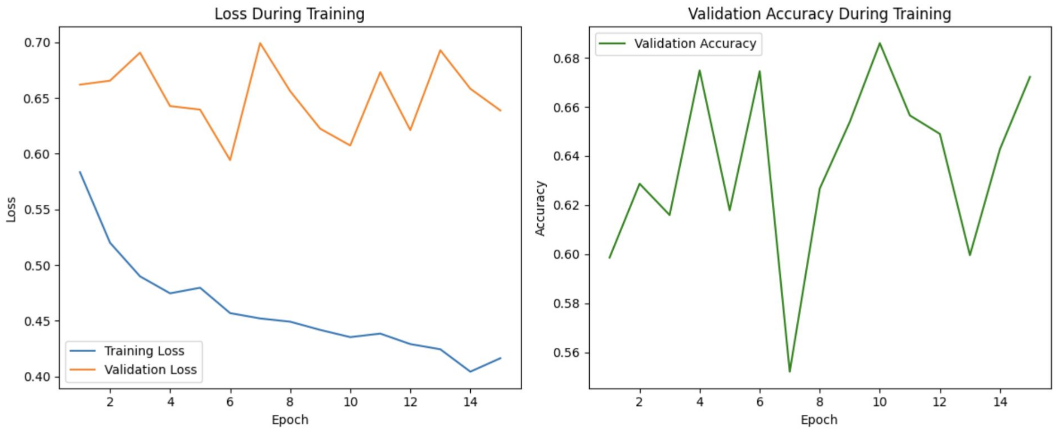
Similar performance to variant 2. However, the important point that stands out is the apparent lower volatility of validation performance

Variant 3: Sox10



```
=== Test Set Performance ===
Accuracy     : 0.7796
Precision    : 0.9057
Recall       : 0.8105
F1 Score     : 0.8555

Classification Report:
                 precision    recall  f1-score   support

         Benign       0.45      0.65      0.54       666
High-grade CMIL       0.91      0.81      0.86      2750

       accuracy                           0.78      3416
      macro avg       0.68      0.73      0.70      3416
   weighted avg       0.82      0.78      0.79      3416
```





Marginal improvement over variant 2, but otherwise roughly the same. Sox10 continues to deliver best performance

Variant 3 Comments

- Increasing batch size definitely seems to have helped
- First variant to include data augmentation, which could have played a part in some of the differences observed
- Further increasing batch size seems to be the way to go, assuming RAM overload problems do not persist

Final Takeaways

- Sox10 seems to be a great starting point for evaluating validity of different model architectures, delivering best performance for all 3 variations
- Resize versus adaptive pooling: resizing better pick for now, but could definitely revisit adaptive pooling at some point (especially if modifying the layer from original ResNet50 network)
- Continue increasing batch size (attempt 16 and 32 if possible) and see impact on gradient smoothing
- Introduce CBAM to allow model to focus on more relevant components of patches, which could allow improvements in h&e and melan stains
- Unfreezing more layers of the base network may not be very feasible given our computational constraints
- Training single model on all 3 stains