AN2DL - Second Challenge Report
ANeuronInFour

Carmen Giaccotto, Davide Bertoni, Simone Pio Bottaro, Francesco Lauria

carmengiaccotto, *davidebertoni*, *simonepiobottaro*, *fralauria*

286745, 300380, 286706, 252766

December 16, 2025

1 Introduction

This project addresses the challenge of *molecular subtype classification* within human tissue using images derived from low-magnification Whole Slide Imaging (WSI). The primary **goal** is to develop a robust predictive model capable of accurately classifying images into one of four possible classes: **Luminal A, Luminal B, HER2(+)**, or **Triple Negative** (Fig. 1). Our approach will involve applying and optimizing **deep learning techniques** to extract relevant features and build a highly accurate classification system.

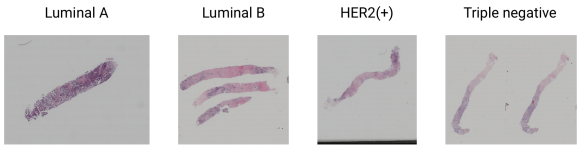

Figure 1: Molecular subtypes

2 Problem Analysis

2.1 Dataset Characteristics

The dataset contains **1168 Image/Mask** pairs for molecular subtype classification:

- **Training set**: images with corresponding binary masks (regions where the disease is located) 2. Labels are provided in *train_labels.csv*.

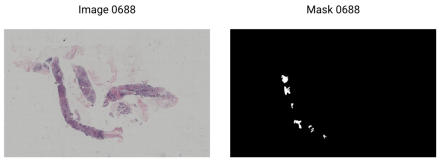

Figure 2: Image with the relative auxiliary mask

- **Test Set**: unlabeled images/masks for final evaluation.

2.2 Main Challenges

In order to develop a robust image classifier, we needed to address some challenges related to the dataset:

1. We preemptively identified and removed a list of **110 images** from the training set with the corresponding auxiliary masks. 50 of them contained a green stain (as visible in Figure 3) and were duplicates of other images already present in the training set. The remaining 60 did not represent valid training examples.

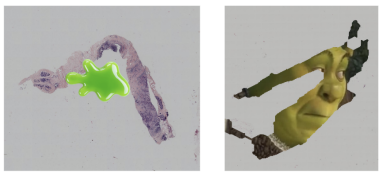

Figure 3: Examples of removed Images

2. **Data scarcity**: the dataset is very small, expecially the images were quite noisy, so we had to apply many techniques to artificially augment the amount of data to be fed to the CNN, such as the Grid Tiling Strategy and heavy **augmentation** on the tranforms with pytorch.

3. To handle high dimensional variability and focus on tumor tissue, we employed a **Grid Tiling strategy**. This technique extracts fixed 224×224 patches with 30% overlap from the images. Only patches that intersect the binary mask by at least 2% are kept. Areas outside the tissue are filled with white pixels (standard background) to maintain visual consistency.

4. To ensure chromatic consistency and optimize input distribution for the CNN, we implemented a two-step normalization pipeline. First, we applied **Macenko Normalization** [4] using reference stain vectors (median HE and maxC) derived from a random sample of 100 WSIs (non-tumor regions were masked prior to calculation). Subsequently, we performed **Statistical Normalization**, recomputing the channel-wise mean and standard deviation on a random 20% subset of the Macenko-normalized training data.

5. To mitigate the prominent **class imbalance** (Figure 4), we employed two main methods. We used Stratified K-Fold cross-validation to preserve class ratios across data partitions. Additionally, we penalized minority classes misclassifications by incorporating dynamic class weights (calculated per fold) into a Focal Loss function.

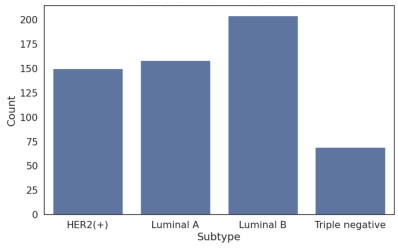

Figure 4: Class Distribution in Full Dataset

3 Method

We implemented Transfer Learning and Fine Tuning with cross-validation strategy to maximize generalization. We selected **EfficientNet** [3] as the backbone architecture due to its strong balance of parameter efficiency and performance on complex image data. We used the **Lion optimizer** for training.

To increase model's robustness, especially during the Fine-Tuning phase, we applied several **data augmentation** strategies. The pipeline consisted of two main types of transformations: **Spatial Invariance Transformations** (random resizing, cropping, horizontal/vertical flips, and random rotations), and **Elastic Transformations** to simulate realistic tissue distortions. Additionally, **Photometric Invariance Transformations** were applied at the pixel level to enhance resilience against staining variations, specifically random variations in brightness, contrast, saturation, and hue (`ColorJitter`), and mild Gaussian blurring (`GaussianBlur`). Furthermore, we employed **MixUp** as an advanced regularization technique during Fine-Tuning. This approach generates synthetic training data $(\tilde{x}, \tilde{y})$ by forming a convex combination of two random training examples (x_i, y_i) and (x_j, y_j): $\tilde{x} = \lambda x_i + (1 - \lambda)x_j$ and $\tilde{y} = \lambda y_i + (1 - \lambda)y_j$, where λ is sampled from Beta(α, α) with $\alpha = 0.2$ (`MIXUP_ALPHA`).
We employed a **6-Fold Stratified Cross-Validation ($K = 6$) strategy**. This approach trained five independent EfficientNet models, with each model trained on 83.3% of the data and validated on the remaining 16.7%.

4 Experiments

4.1 One-vs-Rest (OvR) Classification Stacking

We tried implementing a completely different strategy and architecture to solve the classification task: **One-vs-Rest (OvR) classification stacking architecture**, a method that has shown promising results in a similar research [5]. The objective was to reduce the problem to four binary classification problems, allowing specialized models to better differentiate each molecular subtype from the remain-

ing classes. Then ensembling the results with an XGboost classifier [2]. Despite its demonstrated effectiveness in published literature , this approach did not yield a superior F1-Score compared to our direct K-Fold ensemble method.

4.2 Xtreme Gradient Boosting

One approach we tested involved using **eXtreme Gradient Boosting (XGBoost)** as a second-stage classifier. The core idea was to use the soft voting probabilities generated by the 6 K-Fold models as features to train the XGBoost model. However the results obtained did not show a significant improvement over the performance achieved by the simple ensemble voting.

4.3 Optimizers

We compared the **AdamW** optimizer with the **Lion** (Lookahead Optimizer with Implicit Gradients) algorithm. Although AdamW proved promising in initial tests, we found that the Lion configuration consistently delivered the best results and the highest final classification metrics across all cross-validation folds. For this reason, Lion was selected as the final optimizer for the complete pipeline.

4.4 LR Scheduler

We tried using both **ReduceOnPlateau** and **CosineAnnealing** learning rate schedulers, we chose the first as it was providing more consistent results.

5 Discussion and Results

The primary strength of our work lies in the robustness and quality of the data pipeline, ensuring meticulous preparation for training. However, we encountered a critical limitation: the inability to significantly reduce the validation loss beyond a certain threshold, despite exhaustive exploration of various architectures and optimizers. This plateauing led to lower generalization performance on unseen data. To identify the most suitable model for our classification task, we analyzed several cutting-edge architectures: **EfficientNet, ResNet, PhiCoN**, and **the specialized UniMahmood** [1]. Table 1 summarizes the best results achieved during Inference. The EfficientNet architecture achieved the best overall result, making it the final choice for the ensemble submission.

Table 1: Models explored. Best results are highlighted in **bold**.

Model	F1 on Inference (%)
EfficientNet	**40.52**
ResNet	30.09 (Old Dataset)
PhiCoN	38.93
UniMahmood	40.41
MobileNet	39.19

5.1 Grad-CAM Analysis

To gain deeper insight into the models' decision-making and validate the success of the ROI Tiling strategy, we performed **Gradient-weighted Class Activation Mapping (Grad-CAM)** visualizations. Grad-CAM produces a coarse localization map highlighting the regions in the image that were most influential for the model's final classification decision. The results visually confirmed that the model was focusing its attention primarily on the regions of biological tissue contained within the extracted patches (Figure 5).

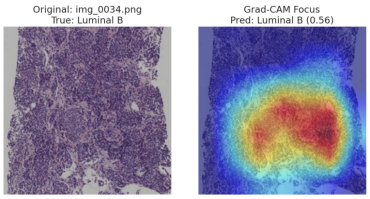

Figure 5: Grad CAM Focus example

6 Conclusions

While the results are encouraging, the persistence of the validation loss plateau indicates that further improvement is certainly achievable. Moving forward we should focus on optimizing model performance beyond the current threshold by conducting a more extensive search for optimal hyperparameters, especially those governing the Lion optimizer and the Learning Rate Scheduler. Additionally, we must investigate the effect of adding more complex classifier layers (heads) atop the different backbone architectures and implement advanced augmentation policies like RandAugment and AugMix to further increase data variability and model generalization.

References

[1] D. T. L. M. W. D. Chen, R.J. Towards a general-purpose foundation model for computational pathology, 2024. https://doi.org/10.1038/s41591-024-02857-3.

[2] G. C. Chen T. Xgboost: a scalable tree boosting. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*, 2016.

[3] P. Foundation. Models and pre-trained weights. https://docs.pytorch.org/vision/main/models.html.

[4] GeeksforGeeks. Macenko method for normalizing histology slides for quantitative analysis. https://www.geeksforgeeks.org/machine-learning/macenko-method-for-normalizing-histology-slides-for-quantitative-analysis/, 2025.

[5] R. M. S. N. B. L. B. L. M. K. Tafavvoghi M, Sildnes A. Deep learning-based classification of breast cancer molecular subtypes from he whole-slide images. *Journal of Pathology Informatics*, 2024.