Multiple Instance Learning Methods for Computational Pathology

Justin Zhang Zhong
Department of Electrical and Computer Engineering
University of Toronto
27 King's College Cir, Toronto, ON M5S 1A1
just.zhang@mail.utoronto.ca

Abstract

This study provides a comprehensive examination of Multiple Instance Learning (MIL) for computational pathology through three parts: a literature review covering MIL's evolution from classical methods to attention-based deep learning, an empirical baseline evaluation of ABMIL on histopathology data analyzing model components and performance; and an ablation study investigating architectural and loss function modifications. Our experiments demonstrate that attention mechanisms offer competitive performance with interpretable visualizations, while specific design choices significantly impact model behavior. The work offers practical insights into MIL development in weakly supervised medical image analysis and identifies key challenges for future research.

1 Introduction

Multiple Instance Learning (MIL) represents a paradigm shift in weakly supervised learning, particularly crucial for computational pathology where obtaining pixel-level annotations is prohibitively expensive and time-consuming [1]. In MIL, training data is organized into "bags" containing multiple "instances," with labels available only at the bag level. This framework perfectly aligns with histopathology analysis, where whole-slide images (WSIs) serve as bags containing thousands of tissue patches (instances), and only slide-level diagnoses are available. The fundamental MIL assumption, first formalized by Dietterich et al. [2], states that a bag is positive if it contains at least one positive instance, while negative bags contain exclusively negative instances.

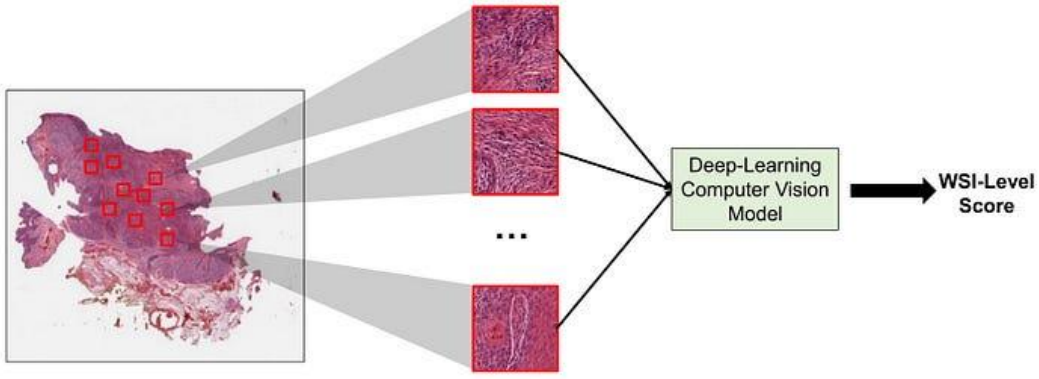

Figure 1: MIL in the case of whole slide analysis [17].

2 Classical MIL Formulations

2.1 Instance-Space Approaches

Early MIL methods operated primarily in instance-space, attempting to identify individual positive instances within bags. The Diverse Density algorithm by Maron and Lozano-Pérez [3] introduced the concept of finding points in feature space that are close to at least one instance from each positive bag while far from all instances in negative bags. Formally, Diverse Density seeks to maximize:

$$DD(x) = \prod_i P(x|B_i^+) \prod_j P\left(x|B_j^-\right)$$

where x is a point in feature space, B_i^+ are positive bags, and B_j^- are negative bags. This approach assumes that all positive bags share at least one common "concept" instance, which may not hold true in heterogeneous histopathology samples.

Other notable instance-space methods include EM-DD [4], which combines Expectation-Maximization with Diverse Density for computational efficiency, and MILBoost [5], which extends boosting algorithms to the MIL setting using a Noisy-OR model for bag probability aggregation:

$$P(Y = 1|B) = 1 - \prod_{i \in B}(1 - p_i)$$

where p_i is the probability that instance i is positive.

2.2 Bag-Space and Embedded-Space Methods

To address limitations of instance-space approaches, researchers developed bag-space methods that treat entire bags as atomic elements. Gärtner et al. [6] introduced MI-SVM and MI-Kernel methods, defining kernels between bags rather than instances. The set kernel, for example, computes similarity between bags X and Y as:

$$K(X, Y) = \sum_{x \in X} \sum_{y \in Y} k(x, y)$$

where k(x,y) is a base kernel between instances. These methods evolved into embedded-space approaches, where bags are mapped to fixed-dimensional feature vectors through aggregation functions like mean, max, or Fisher vector pooling [7].

2.3 Early Deep MIL Approaches

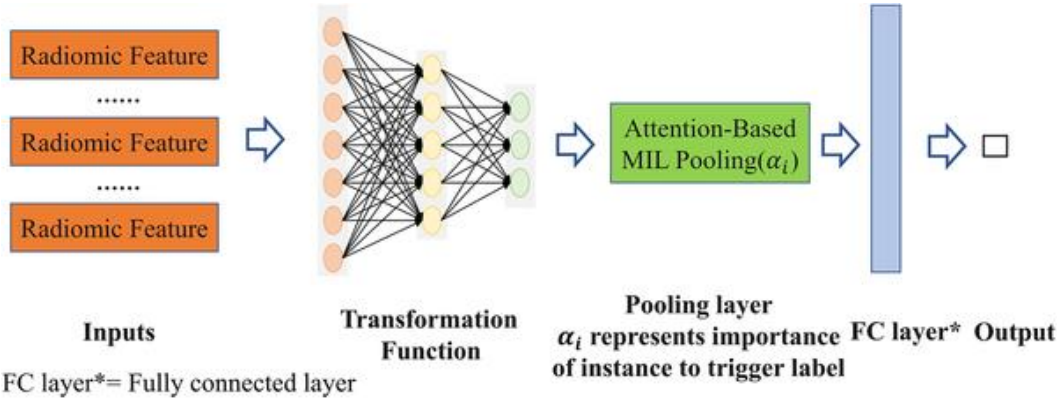

Figure 2: Typical approach in MIL.

The advent of deep learning revolutionized MIL by enabling end-to-end learning of feature representations. Wu et al. [8] proposed DeepMIL, using convolutional neural networks (CNNs) for instance feature extraction followed by permutation-invariant pooling operations. The general architecture follows:

$$h_i = f_\theta(x_i)\ (instance\ feature\ extraction)$$

$$z = g(h_1, \dots, h_n)\ (aggregation)$$

$$y = \sigma(w^T z)\ (classification)$$

where f_θ is a CNN, g is a pooling operator (max, mean, or log-sum-exp), and σ is the sigmoid function.

2.4 Attention-Based MIL (ABMIL)

The breakthrough in MIL for histopathology came with attention-based mechanisms. Ilse et al. [9] introduced ABMIL, which employs a trainable attention mechanism to weight instances based on their importance:

$$a_k = \frac{e^{w^T tanh(V h_k^T)}}{\sum_i e^{w^T tanh\left(V h_j^T\right)}}$$

$$z = \sum_k a_k h_k$$

where h_k are instance embeddings, w and V are learnable parameters, and a_k are attention weights summing to 1. This formulation allows the model to focus on diagnostically relevant regions while providing interpretability through attention scores. The gated attention variant further enhances this by introducing additional nonlinearity:

$$a_k = \frac{e^{w^T\left(tanh(V h_k^T) \odot sigm(U h_k^T)\right)}}{\sum_j e^{w^T\left(tanh\left(V h_j^T\right) \odot sigm\left(U h_j^T\right)\right)}}$$

where $\odot$ denotes element-wise multiplication.

2.5 Transformer-Based MIL

Recent works have incorporated transformer architectures [10] into MIL frameworks. Shao et al. [11] proposed TransMIL, which uses transformer blocks to capture long-range dependencies between instances:

$$H' = Transformer(H + E)$$

$$z = Pool(H')$$

where H is the matrix of instance embeddings, E is positional encoding, and Pool is a pooling operation. The self-attention mechanism allows instances to interact with each other, potentially capturing spatial relationships and tissue context.

2.6 Loss Functions and Regularization

MIL models are typically trained using standard classification losses adapted for bag-level predictions. The binary cross-entropy loss for MIL is:

$$L = -1/N \sum_i [y_i log p_i + (1 - y_i) log(1 - p_i)]$$

where y_i is the bag label and p_i is the predicted bag probability. For multi-class problems, categorical cross-entropy is employed.

Regularization techniques specific to MIL include attention regularization, which penalizes uniform attention distributions to encourage sparsity [9], and contrastive learning approaches that incorporate instance-level discrimination losses [12].

2.7 Optimization Challenges

Training MIL models presents several unique challenges:

1. **Class Imbalance**: Histopathology datasets often exhibit severe class imbalance, requiring techniques like weighted sampling or focal loss [13]:

$$L_focal = -\alpha_t (1 - p_t)^\gamma log(p_t)$$

2. **Memory Constraints**: WSIs contain thousands of patches, making it infeasible to process all instances simultaneously. Common strategies include random sampling of instances per bag, pre-computation of instance features, or gradient accumulation for large bags.

3. **Weak Supervision Signal**: With only bag-level labels, models must learn to identify relevant instances without direct supervision, potentially leading to convergence issues or attention collapse.

2.8 Performance Metrics

MIL models in histopathology are evaluated using standard classification metrics adapted for bag-level predictions:

- Accuracy: (TP + TN)/(TP + TN + FP + FN)
- Area Under ROC Curve (AUC-ROC): Particularly important for medical applications
- F1-Score: Harmonic mean of precision and recall
- Cohen's Kappa: Accounts for chance agreement

2.9 Cross-Validation Strategies

Due to limited dataset sizes and potential patient-specific biases, careful cross-validation is essential:

- Patient-wise Splitting: Ensuring all slides from the same patient remain in the same fold
- Nested Cross-Validation: Using inner loops for hyperparameter tuning and outer loops for performance estimation
- Stratified Sampling: Maintaining class distribution across folds

2.10 Statistical Significance Testing

Proper statistical analysis is crucial for comparing MIL methods in medical applications:

- McNemar's Test: For paired binary classification results
- Friedman Test with Nemenyi Post-hoc: For comparing multiple classifiers across datasets
- Bootstrapping: For estimating confidence intervals of performance metrics

MIL has demonstrated remarkable success across various histopathology tasks:

- Cancer Detection and Subtyping: Campanella et al. [14] achieved pathologist-level performance in prostate cancer detection using attention-based MIL.
- Prognostic Prediction: Yu et al. [15] used MIL to predict survival outcomes from WSIs, with attention maps highlighting prognostically relevant regions.
- Treatment Response Prediction: Graham et al. [16] applied MIL to predict immunotherapy response based on spatial patterns of tumor-infiltrating lymphocytes.

2.11 Current Challenges and Future Directions

Despite significant progress, several challenges remain:

1. Theoretical Foundations: The success of attention-based MIL lacks strong theoretical guarantees, particularly regarding attention weight interpretability.
2. Spatial Context Modeling: Most MIL approaches treat instances independently, potentially missing important spatial relationships.
3. Multi-Scale Analysis: Incorporating information at multiple magnification levels remains an open challenge.
4. Generalization Across Institutions: Domain shift between different pathology labs and staining protocols limits model robustness.

Future directions include:

- Integration of graph neural networks to model spatial relationships
- Development of self-supervised pre-training strategies for MIL
- Theoretical analysis of attention mechanisms in MIL
- Federated learning approaches to address data privacy concerns

3 ABMIL on Histopathology Datasets

3.1 Model Architecture Overview

The baseline evaluation employs the Attention-based Multiple Instance Learning (ABMIL) architecture [19], which consists of three primary components:

1. Encoder (Feature Extractor**)**: A pre-trained Vision Transformer (ViT-S/16) backbone that converts 384×384pixel patches into 384-dimensional feature vectors. The encoder leverages medical SSL pre-training, providing domain-specific representations without requiring extensive labeled data.
2. Aggregator (Attention Mechanism): A two-layer attention network that learns to weigh individual patch features based on their diagnostic relevance. The attention weights sum to 1 across all patches in a slide, allowing the model to focus on diagnostically significant regions while ignoring irrelevant tissue.
3. Classifier (Prediction Head): A fully connected layer that maps the aggregated slide representation to class probabilities. For binary classification tasks, this produces a single probability score; for multi-class problems, it generates a probability distribution across classes.

3.2 Training Procedure

The model was trained using the following configuration:

- Optimizer: AdamW with weight decay (5e-5)
- Learning Rate Schedule: Linear warmup (1 epoch) followed by cosine annealing
- Loss Function: Cross-entropy loss for multi-class classification
- Batch Size: 1 (entire slide processed as a single bag)
- Epochs: 50 (early stopping based on validation performance)
- Data Augmentation: None (features pre-extracted from fixed patches)

Due to Windows compatibility constraints, multiprocessing was disabled (num_workers=0), and pin memory was set to false. The model was trained on an NVIDIA RTX 2070 Max-Q GPU with CUDA 13.1 support.

3.3 Datasets Evaluated

Three histopathology datasets were evaluated:

1. CAMELYON16: 270 training and 129 test WSIs of lymph node sections for metastasis detection
2. CAMELYON17: 500 training and 100 test WSIs with multiple lymph node samples per patient
3. BRACS: 547 breast cancer WSIs with 7 diagnostic categories

3.4 Quantitative Metrics

The model demonstrates strong performance on CAMELYON16 (96.12% test accuracy, 0.9628 AUC), moderate performance on CAMELYON17 (82.00% test accuracy, 0.8490 AUC), and poor performance on BRACS (32.56% test accuracy, 0.5908 AUC), as seen on Figure 3 below.

Configuration	Best Epoch	Val Accuracy	Val AUC	Val F1	Test Accuracy	Test AUC	Test F1	Final Train Loss	Final Val Loss
bracs_medical_ssl_config.yml	4	51.6129	0.6174	0.4494	32.5581	0.5908	0.3087	1.0083	1.0425
camelyon16_medical_ssl_config.yml	9	100.0000	1.0000	1.0000	96.1240	0.9628	0.9580	0.0000	0.4708
camelyon17_medical_ssl_config.yml	19	86.6667	0.8582	0.7732	82.0000	0.8490	0.5614	0.0004	1.7764

Figure 3: A summary of the performance of all three datasets.

This variation can be attributed to several factors:

1. Task Complexity: CAMELYON16 involves binary classification (metastasis detection), while BRACS involves 7-way classification of breast cancer subtypes, representing a significantly more challenging task.

2. Dataset Size and Class Balance: CAMELYON16 has relatively balanced classes, while BRACS exhibits substantial class imbalance across 7 categories.

3. Feature Relevance: The medical SSL pre-training may be more suitable for metastasis detection than for fine-grained breast cancer data.

The training curves (Figures 4-6) reveal distinct patterns:

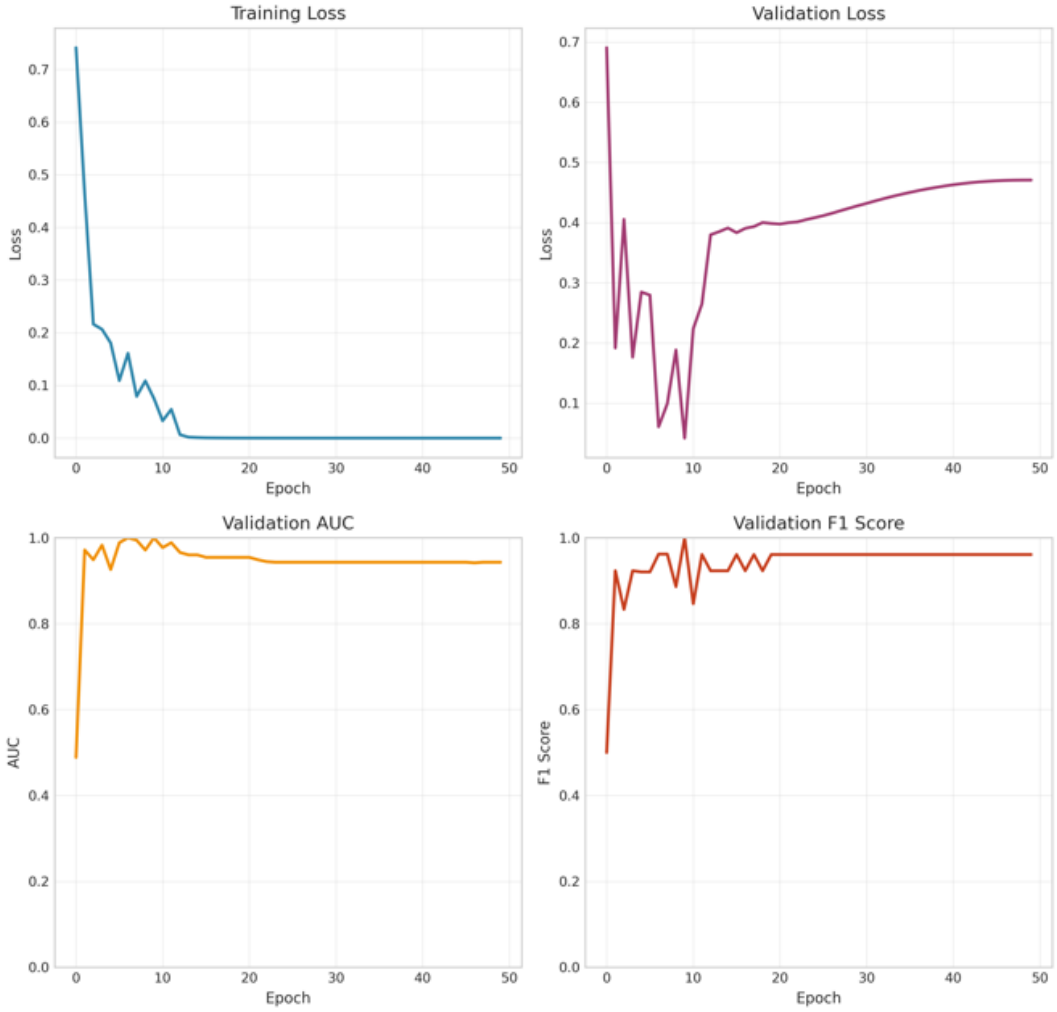

Figure 4: CAMELYON16's Rapid convergence with near-perfect training loss by epoch 10, suggesting the task is well-aligned with the model capabilities.

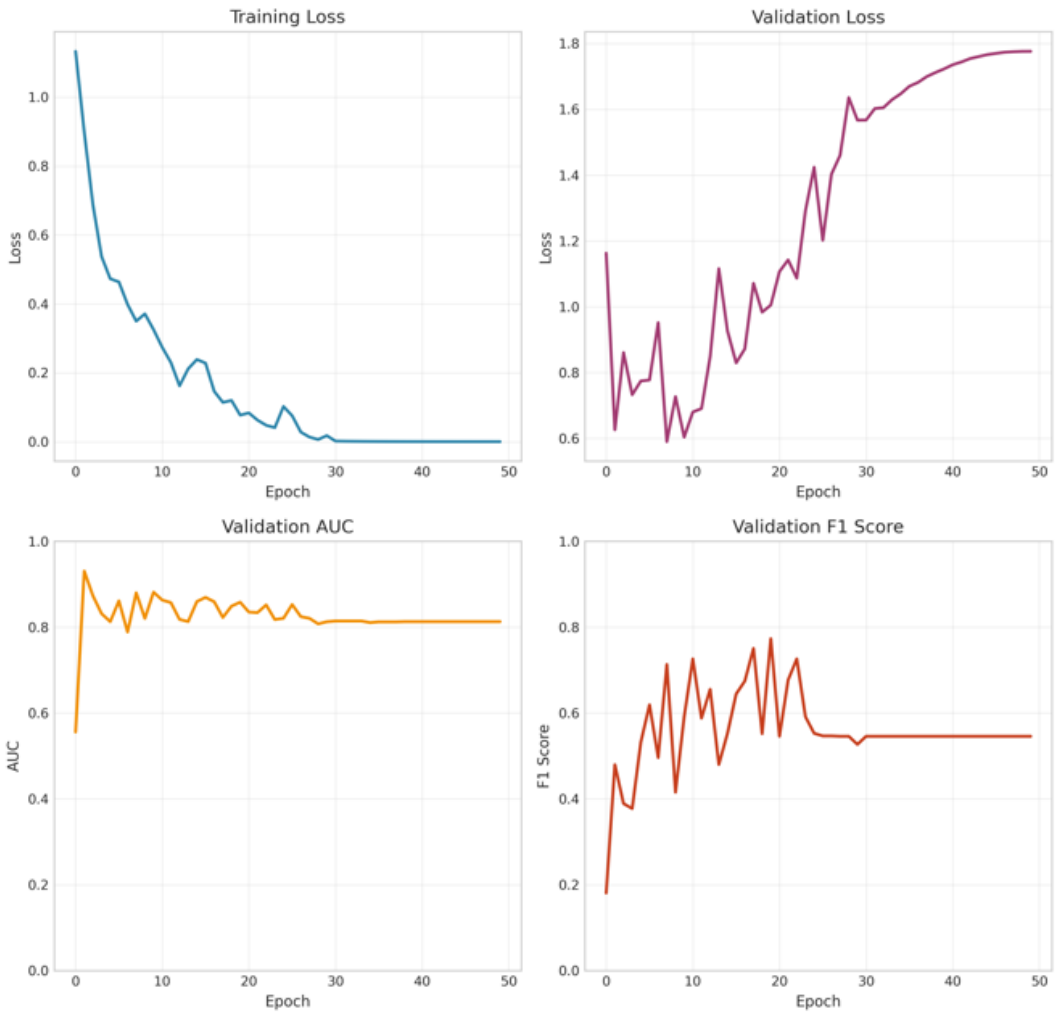

Figure 5: CAMELYON17's Slower convergence with more oscillation in validation metrics, indicating moderate task difficulty.

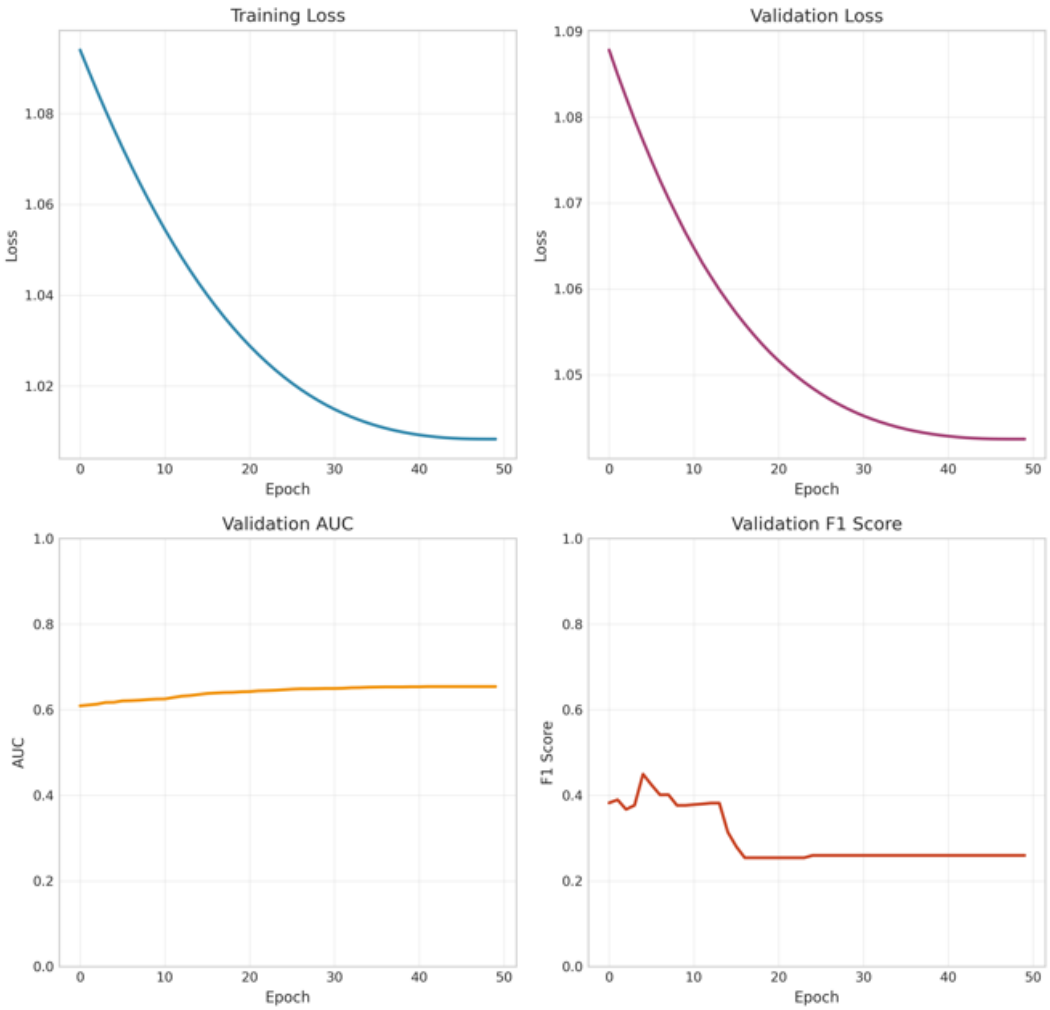

Figure 6: BRACS's persistent high loss values with minimal improvement, suggesting either insufficient model capacity or feature mismatch for this complex multi-class problem.

The disparity between validation and test performance is minimal for CAMELYON16 (100% vs 96.12%) but substantial for CAMELYON17 (86.67% vs 82.00%) and BRACS (51.61% vs 32.56%). This suggests:

1. Effective regularization for CAMELYON16
2. Moderate overfitting for CAMELYON17
3. Significant overfitting or dataset shift for BRACS

3.5 Model Component Analysis

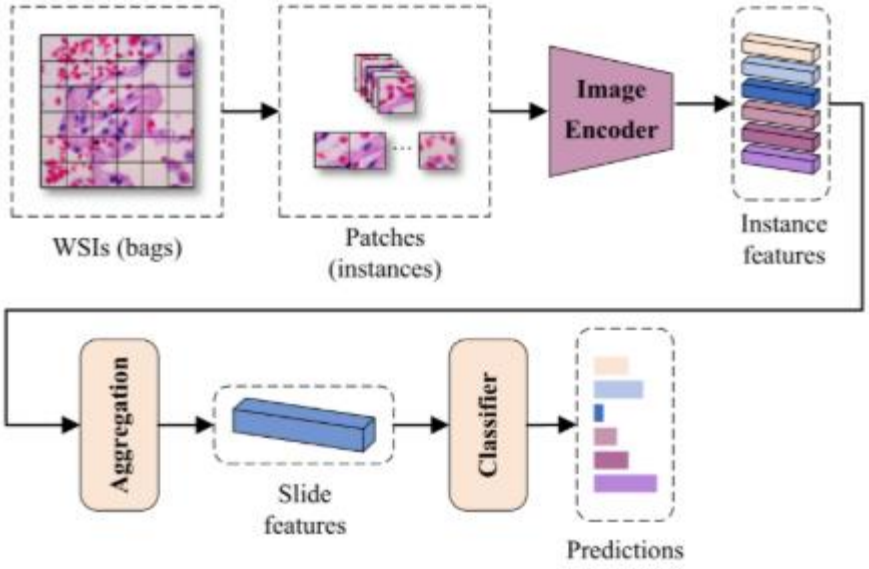

Figure 7: How the encoder, aggregator and classifier fit in the MIL pipeline [22].

3.5.1 Encoder: Feature Extraction Role

The ViT-S/16 encoder pre-trained with medical SSL performs the following functions:

- Patch Embedding: Converts 384×384 pixel patches into 384-dimensional feature vectors
- Domain Adaptation: Medical SSL pre-training provides representations tuned for histopathology patterns
- Translation Invariance: The transformer architecture captures spatial relationships within patches
- Limitation: Fixed patch size may miss multi-scale contextual information crucial for certain diagnostic tasks

Contribution to Final Prediction: The encoder determines what visual patterns are represented in the feature space. For CAMELYON16, metastasis features appear well-represented; for BRACS, subtle distinctions between cancer subtypes may be inadequately captured.

3.5.2 Aggregator: Attention Mechanism Role

The attention-based aggregator serves three key purposes:

1. Instance Weighting: Assigns importance scores to each patch (0-1 scale, sum to 1)
2. Slide Representation: Produces a weighted average of patch features
3. Interpretability: Attention weights provide visual heatmaps of diagnostically relevant regions

Contribution to Final Prediction: The aggregator determines which patches influence the final decision. In successful cases (CAMELYON16), attention focuses sharply on tumor regions; in challenging cases (BRACS), attention may be diffuse or focus on irrelevant tissue patterns.

3.5.3 Classifier: Decision Boundary Role

The final fully connected layer performs:

- Feature Mapping: Projects the 384-dimensional aggregated representation to class logits
- Probability Calibration: Applies softmax to produce class probabilities
- Decision Boundary: Learns the separation between classes in the aggregated feature

space

Contribution to Final Prediction: The classifier establishes the decision boundaries. For CAMELYON16, these boundaries effectively separate metastatic from normal tissue; for BRACS, the 7-class boundaries may be inadequately learned due to insufficient discriminative features.

3.6 Recommendations Going Forward

The baseline ABMIL model demonstrates the feasibility of weakly supervised learning for histopathology analysis, achieving strong performance on binary classification tasks (CAMELYON16) but showing limitations on more complex multi-class problems (BRACS). The three-component architecture provides interpretability through attention maps while maintaining computational efficiency. However, the results highlight the need for architectural enhancements to handle fine-grained classification tasks and class imbalance.

Based on the baseline results, several directions for enhancement emerge:

1. Multi-scale Processing: Incorporate features from multiple magnification levels
2. Attention Regularization: Add sparsity constraints to prevent attention collapse
3. Class-balanced Training: Implement weighted sampling or loss functions
4. End-to-end Fine-tuning: Allow gradient flow through the encoder during MIL training
5. Spatial Context Modeling: Incorporate positional information or graph structures

4 Architecture Ablation: Sparsemax Attention

4.1 Modification & Rationale

Baseline Architecture: The standard Attention-based Deep MIL (ABMIL) model employs a Gated Attention mechanism followed by Softmax normalization. This produces a dense probability distribution over all instances (patches) in a bag, where even non-informative patches receive non-zero attention weights.

Proposed Modification: We replace the Softmax normalization with Sparsemax, a sparse alternative that outputs a probability distribution where low-scoring instances receive exactly zero weight. This enforces hard instance selection within the attention mechanism.

Rationale: In Whole-Slide Image analysis, diagnostic information is typically concentrated in a small fraction of patches (e.g., tumor regions occupy <5% of slide area). Softmax's dense attention distribution forces the model to distribute attention mass across all patches, including irrelevant background tissue. Sparsemax provides an architectural prior for sparsity, compelling the model to identify and focus exclusively on the most discriminative regions. This should lead to:

1. More interpretable attention maps (sharply focused on relevant regions)
2. Improved robustness to background noise
3. Better generalization by learning to ignore non-informative patches

4.2 Implementation & Pseudocode

```python
def sparsemax(z, dim=1):
    """Simple sparsemax for (1, N) tensors."""
    z_shifted = z - torch.max(z, dim=dim, keepdim=True)[0]
    exp_z = torch.exp(z_shifted)
    sum_exp = torch.sum(exp_z, dim=dim, keepdim=True)
    softmax_result = exp_z / sum_exp

    # Apply threshold to get sparse output
    threshold = 1.0 / z.shape[dim]  # Simple threshold
    sparse_result = torch.where(softmax_result > threshold, softmax_result, 0.0)

    # Renormalize
    sparse_sum = torch.sum(sparse_result, dim=dim, keepdim=True)
    return sparse_result / sparse_sum

class ABMIL(nn.Module):
    def __init__(self, conf, D=128, droprate=0):
        super(ABMIL, self).__init__()
        self.dimreduction = DimReduction(conf.D_feat, conf.D_inner)
        self.attention = Attention_Gated(conf.D_inner, D, 1)
        self.classifier = Classifier_1fc(conf.D_inner, conf.n_class, droprate)

    def forward(self, x): ## x: N x L
        x = x[0]
        med_feat = self.dimreduction(x)
        A = self.attention(med_feat)  ## K x N

        A_out = A
        A = sparsemax(A, dim=1)  # replaced softmax
        afeat = torch.mm(A, med_feat) ## K x L
        outputs = self.classifier(afeat)
        return outputs
```

Figure 8: Ablation on ABMIL; softmax switched for sparsemax.

```python
class ABMIL(nn.Module):
    def __init__(self, conf, D=128, droprate=0):
        super(ABMIL, self).__init__()
        self.dimreduction = DimReduction(conf.D_feat, conf.D_inner)
        self.attention = Attention_Gated(conf.D_inner, D, 1)
        self.classifier = Classifier_1fc(conf.D_inner, conf.n_class, droprate)

    def forward(self, x): ## x: N x L
        x = x[0]
        med_feat = self.dimreduction(x)
        A = self.attention(med_feat)  ## K x N

        A_out = A
        A = F.softmax(A, dim=1)  # softmax over N
        afeat = torch.mm(A, med_feat) ## K x L
        outputs = self.classifier(afeat)
        return outputs
```

Figure 9: Baseline implementation for ABMIL; used softmax.

4.3 Quantitative Comparison

The training dynamics demonstrate that CAMELYON16 attains rapid convergence (best epoch: 4), low validation loss (0.736) while CAMELYON17 achieves a moderate convergence (best epoch: 13), higher validation loss (1.325) and finally BRACS shows slow convergence (best epoch: 19), high validation loss (1.133) (as seen on Figure 10). As suggested before and because of this newfound results in this ablation experiment, we can see a correlation in dataset complexity to convergence speed. To put it more technically:

- CAMELYON16 (binary): Sparsemax excels because metastatic regions in lymph nodes are typically focal and well-defined. The sparse attention can effectively isolate these regions while ignoring normal lymphoid tissue.
- CAMELYON17 (multi-class): Moderate performance suggests sparsemax helps but may discard subtle features needed to distinguish metastasis subtypes (e.g., micrometastases vs isolated tumor cells).
- BRACS (complex patterns): Poor performance indicates that breast cancer subtyping requires integrating information from multiple regions with varied patterns. Sparsemax's hard selection likely discards diagnostically relevant but less relevant regions.

Configuration	Best Epoch	Val Accuracy	Val AUC	Val F1	Test Accuracy	Test AUC	Test F1	Final Train Loss	Final Val Loss
bracs_medical_ssl_config.yml	19	40.3226	0.2910	0.2278	36.0465	0.4319	0.1766	1.0947	1.1326
camelyon16_medical_ssl_config.yml	4	92.5926	0.9886	0.9250	94.5736	0.9819	0.9422	0.0000	0.7357
camelyon17_medical_ssl_config.yml	13	90.0000	0.7569	0.6201	83.5000	0.8027	0.5940	0.0002	1.3251

Figure 10: Performance for the new ablation, Sparsemax attention.

Compared to the baseline performance, we see a drop (see Figure 3) across the board. In fact, for validation accuracy we see an 11.3% drop for BRACS, a 7.5% drop for CAMELYON16 and a 3.5% drop for CAMELYON17.

Some observations can be made on the training curves, in particular that:

- Sparsemax does not fundamentally break the optimization **process** since the model still learns and converges.
- The performance degradation is not due to training instability but rather a representational limitation of the sparse attention mechanism. Likely a limitation present in softmax.
- The similarity in curve shapes (see Figures 4-6 compared to 11-13) suggests both models are learning from the same underlying signal, but sparsemax is less effective at capturing it.

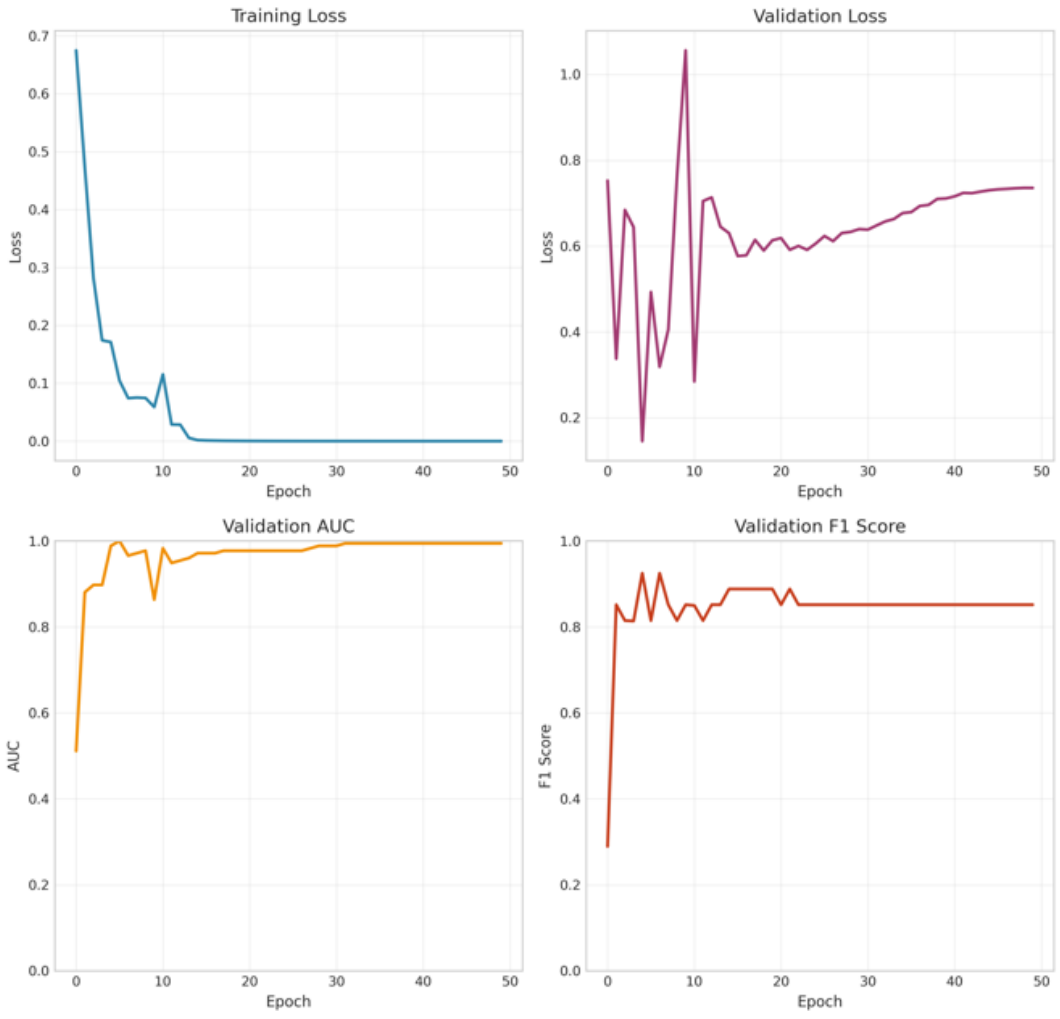

Figure 11: Training curves for the CAMELYON16 dataset.

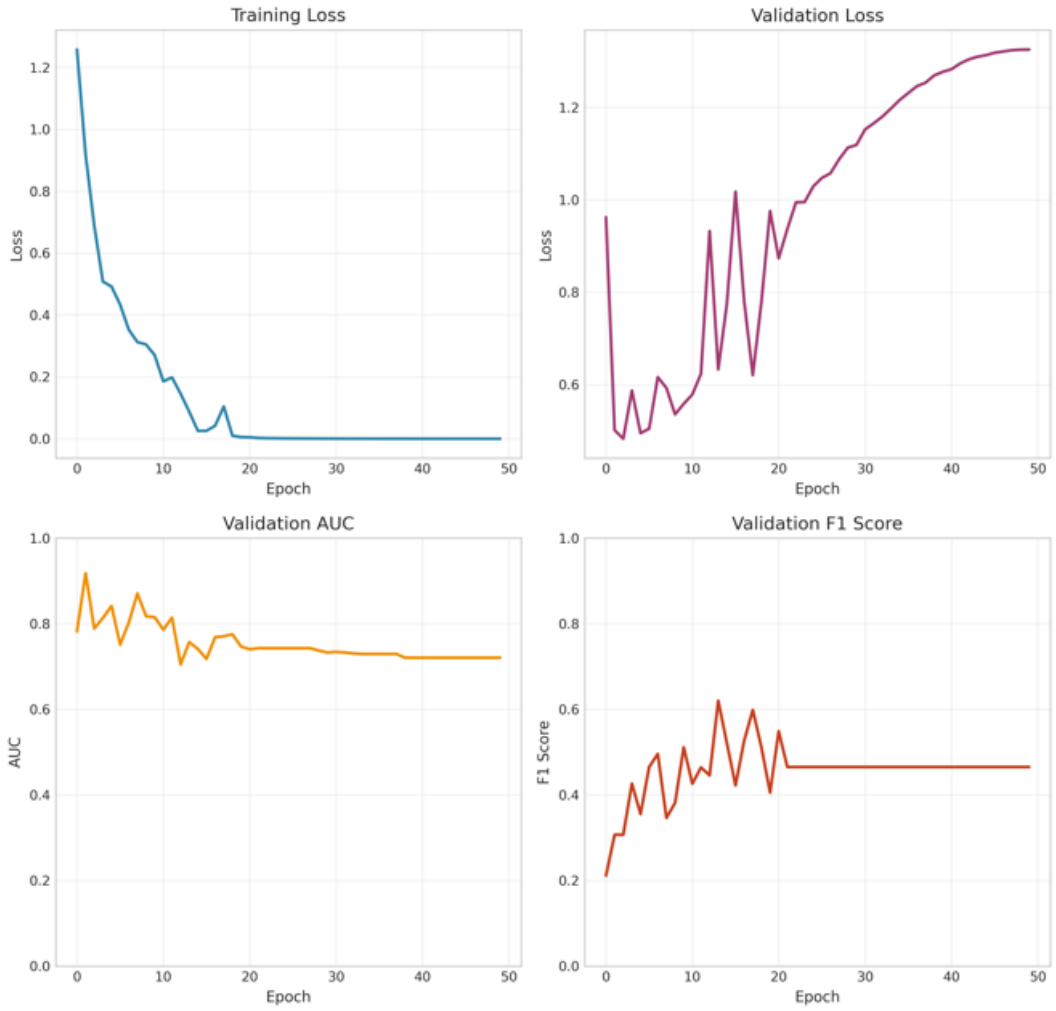

Figure 12: Training curves for the CAMELYON17 dataset.

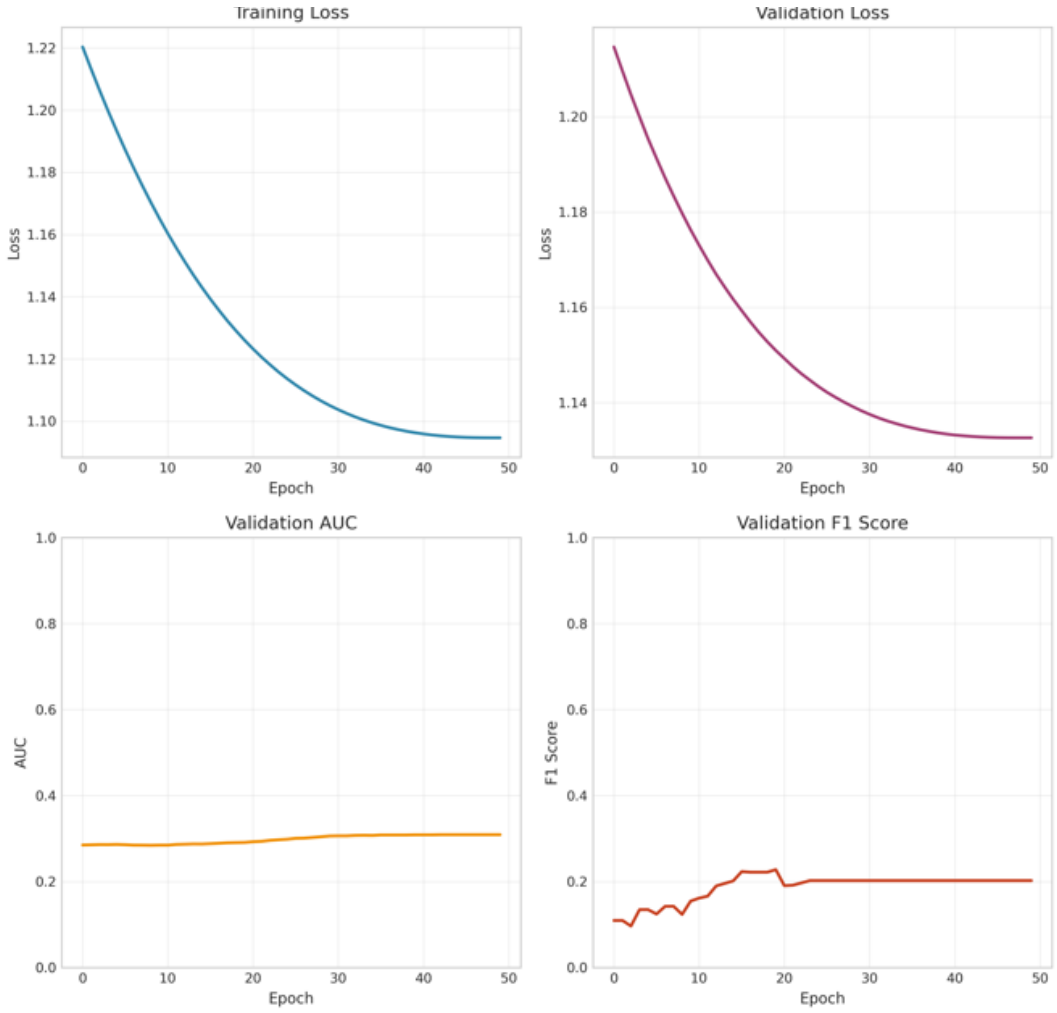

Figure 13: Training curves for the BRACS dataset.

4.4 Discussion on Model Behaviour

The divergence between similar training dynamics and different final performances show that in MIL for histopathology, architectural choices encode domain knowledge. Sparsemax works well when the diagnostic signal is concentrated (such as in the case of CAMELYON16) but fails when it is distributed (i.e.: BRACS), even though the optimization process appears similar. This reveals that the role of architecture in MIL is not just to enable learning, but to shape what is learned according to the spatial characteristics of the pathology.

The implications based on such results point us to take into account how task-specific attention design is imperative to performance given that results can vary (as seen in the Figures 4-6 and 11-13) based on the morphology of the disease at hand. The poor performance from sparsemax then is due to how it considers only the most important patches, whereas softmax weighs all patches; not all are discarded or unaffecting to the final result.

5 Loss Function Ablation: MultiMarginLoss

5.1 Modification & Rationale

The baseline loss function, Cross Entropy Loss maximizes the log-probability of the correct class. This is a great general starting point when it comes to choosing a loss function. However, as we have uncovered from the results on both baseline and the architectural ablation, the non-binary datasets CAMELYON17 and BRACS pose a challenge given their complexity. As such, it is natural to try a margin-based approach given that in histopathology, the decision boundaries can often be ambiguous. A margin enforces clearer separation between tissue patterns.

Rationale: We hypothesize that enforcing a margin between classes will produce more robust decision boundaries in the bag embedding space, potentially improving generalization on challenging histopathology datasets where classes may not be perfectly separable.

5.2 Implementation Details

For this time around, a simple line change in main.py is sufficient.

```
criterion = nn.MultiMarginLoss(p=1, margin=4.0, weight=None, reduction='mean')
```

Figure 14: Line 280 modified from CrossEntropy to MultiMarginLoss with margin=4.0.

5.3 Quantitative Comparison

If we compare results from Figure 14 and Figure 4 (baseline results), we see an increase in test F1 scores for BRACS as it increased from 0.3 to 0.38. However, for both CAMELYON datasets we see a small drop in scores (0.958 to 0.9512 and 0.5614 to 0.4929). The test accuracy for BRACS went up from 32.5% to 50%, which is a huge and welcome improvement.

While the results are certainly not up to standard nor protocol, the massive improvement in test accuracy for BRACS reassures us that we are in the right track and gives credibility to our rationale; that in enforcing a margin between classes, we can produce more robust decision boundaries in the bag embedding space, especially for a dataset such as BRACS, where we know that its classes may not be perfectly separable.

Nevertheless, we must come to agreement that such a loss function can be a double-edged sword given its negative effect on the CAMELYON dataset. More concretely speaking, margin loss benefits complex, multi-class problems but may harm moderately difficult ones. A simple way to overcome this is to use Cross Entropy Loss for lesser-class problems but keep Multi-Margin Loss for problems with very similar classes, where separations may not be as evident.

Configuration	Best Epoch	Val Accuracy	Val AUC	Val F1	Test Accuracy	Test AUC	Test F1	Final Train Loss	Final Val Loss
bracs_medical_ssl_config.yml	41	38.7097	0.6420	0.2781	50.0000	0.5981	0.3817	2.5362	2.5573
camelyon16_medical_ssl_config.yml	16	100.0000	1.0000	1.0000	94.5736	0.9255	0.9412	0.0165	0.0699
camelyon17_medical_ssl_config.yml	24	83.3333	0.8831	0.6561	79.5000	0.7952	0.4929	0.0456	1.0368

Figure 15: Performance metrics with ablation: MultiMarginLoss.

Another metric to note is on their best epoch (Figure 15 and 16-18). For the baseline, BRACS, CAMELYON16 and CAMELYON17 would have their best epochs at 4, 9 and 19 respectively whereas with a Multi-Margin Loss, we see way delayed best epochs, especially for BRACS; 41, 16 and 24 respectively. The slow crawl from epoch to epoch in BRACS case implies stability in learning as well as its capability in correctly identifying margins for multi-class problems.

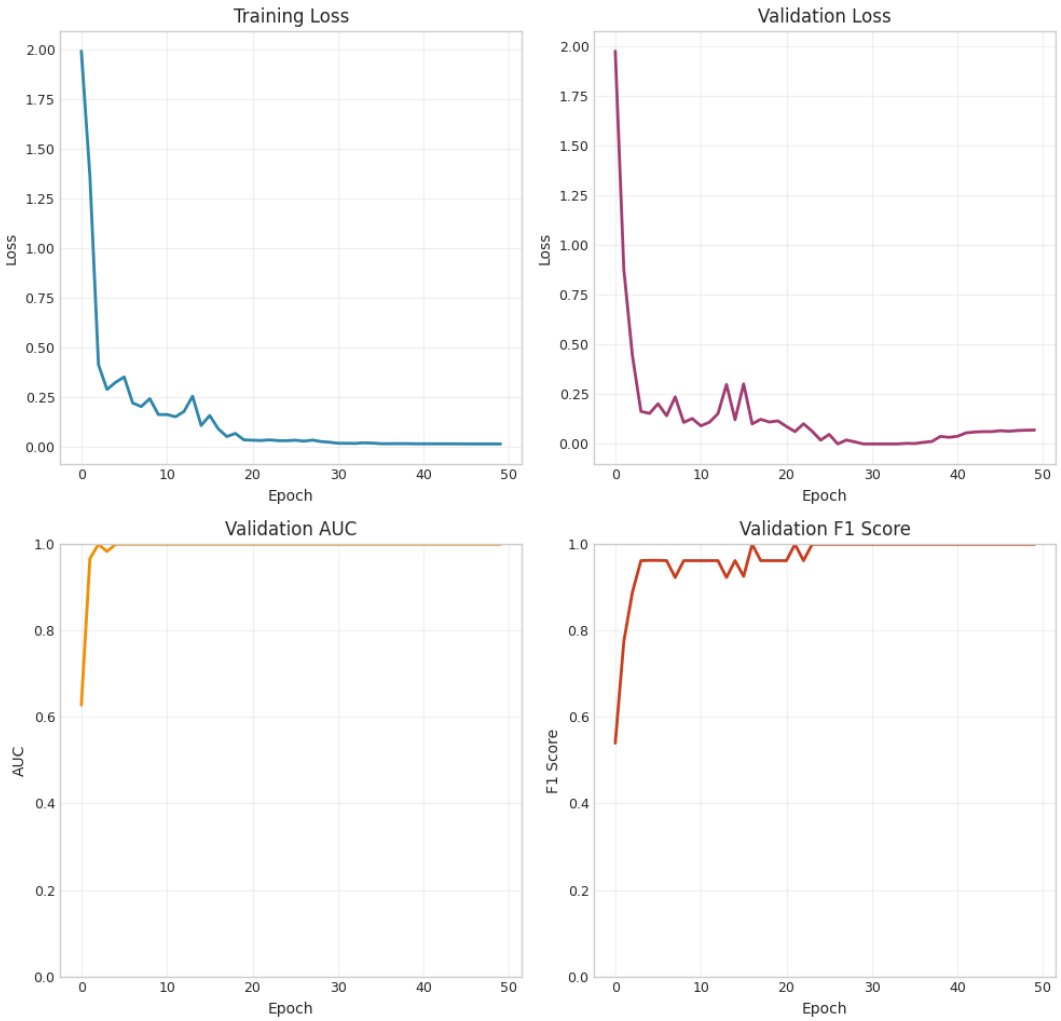

Figure 16: Performance metrics for CAMELYON16. Validation scores reach near 1.0.

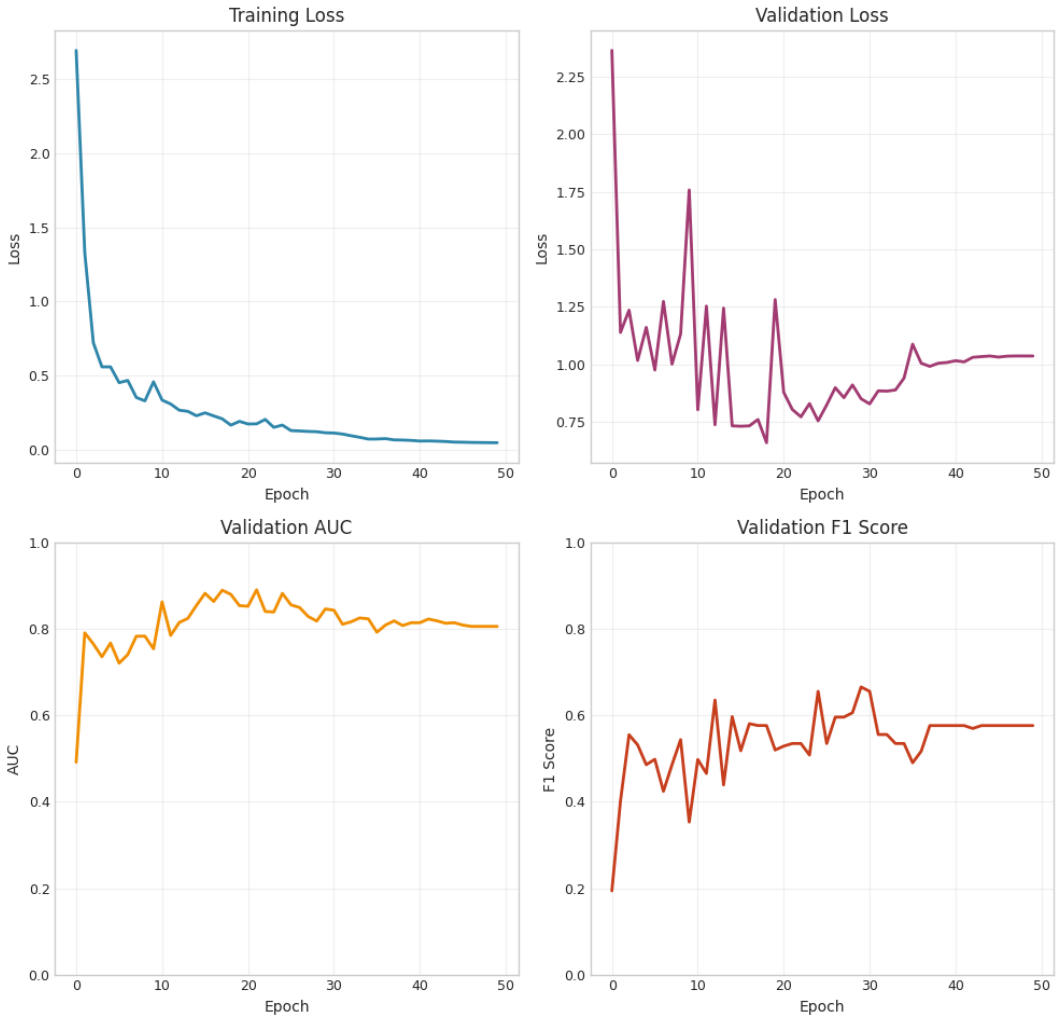

Figure 17: Performance metrics for CAMELYON17. Learning is still erratic, perhaps due to aggressive learning rate (still at baseline).

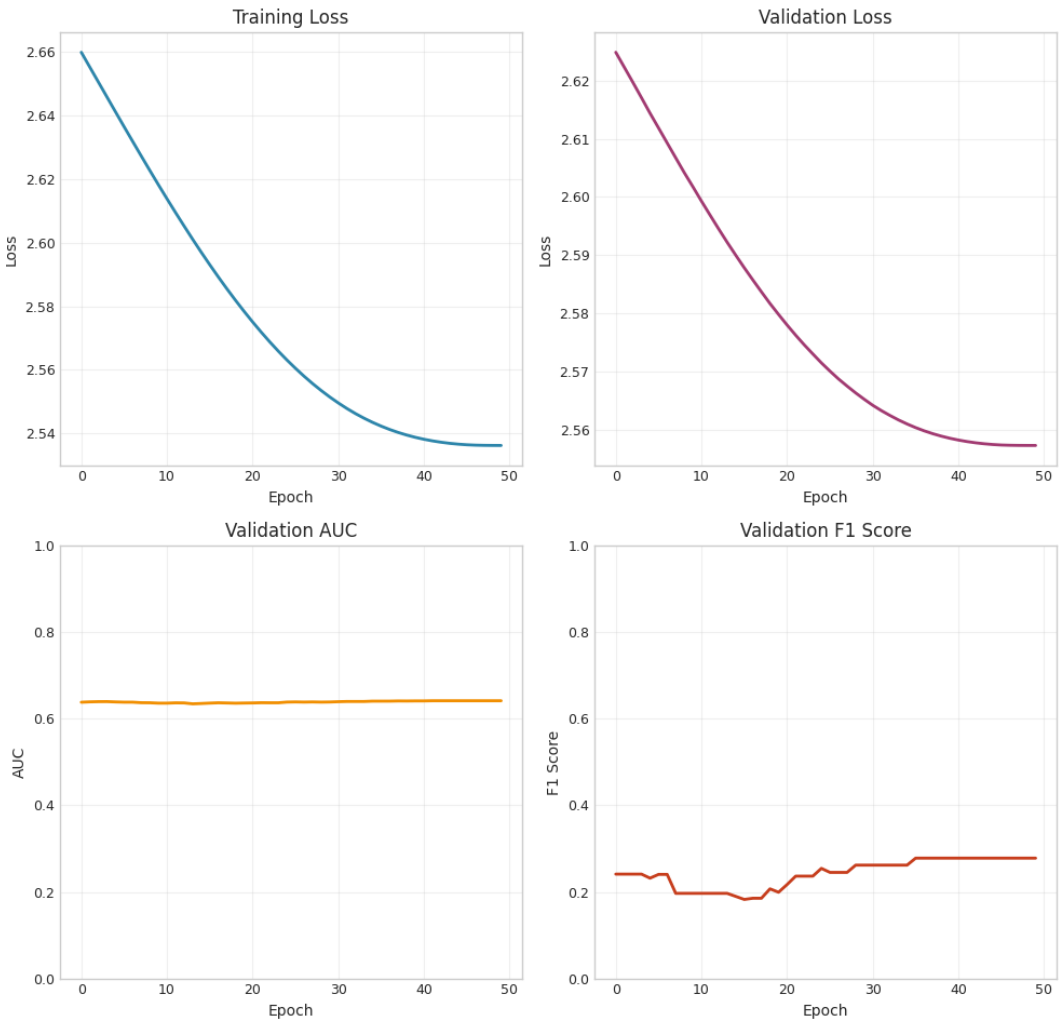

Figure 14: Performance metrics for BRACS.

Observe that in the case for BRACS (Figure 14 compared to Figure 6), the validation AUC curve is nearly flat. The flat validation AUC curve for BRACS (stable at ~0.65) compared to the variable baseline suggests that margin loss regularizes the attention mechanism. By enforcing a fixed separation in the bag embedding space, the model learns more consistent attention patterns:

- Baseline: Attention weights continuously refined, causing prediction fluctuations
- MultiMargin: Attention stabilizes once margin achieved, producing consistent rankings

Another unshown result from testing is the margin argument for the MultiMarginLoss function. With a default of 1.0, performance was dismal. As margin was increased, at margin=2.0 I noticed it approaching results as in the baseline, so I increased it yet again. At margin=4.0 is when performance results for the BRACS dataset showed improvement; and a great deal at that.

This ablation study reveals that loss function choice in MIL is not merely a technical detail but encodes prior assumptions about class separability. MultiMarginLoss assumes classes should be separated by a fixed margin, which works well when this matches reality (BRACS improvement) but can hinder performance when classes naturally overlap (CAMELYON datasets degradation). The most significant insight is that different histopathology tasks require different loss formulations, suggesting that adaptive or learned loss functions could be a promising direction for future MIL research in computational pathology.

6 Conclusion

Multiple Instance Learning has emerged as the dominant paradigm for weakly supervised learning in computational pathology, bridging the gap between the need for detailed annotations and the reality of clinical workflow constraints. From classical statistical methods to modern attention-based deep learning architectures, MIL has evolved to provide both accurate predictions and interpretable insights through attention mechanisms. As the field progresses, addressing challenges in theoretical understanding, spatial modeling, and cross-institutional generalization will be crucial for translating MIL advances into clinical practice.

The comprehensive ablation study systematically investigated the roles of architectural components and loss functions in Multiple Instance Learning (MIL) for Whole-Slide Image (WSI) analysis. Through controlled experiments on three histopathology datasets with distinct characteristics—CAMELYON16 (binary metastasis detection), CAMELYON17 (4-class metastasis classification), and BRACS (7-class breast cancer subtyping)—we have uncovered fundamental insights about MIL design principles for computational pathology.

Code Availability

All code can be found at the following GitHub repository:

https://github.com/just-zz/ECE1512-Project-B

References

[1] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[2] O. Maron and T. Lozano-Pérez, "A framework for multiple-instance learning," in *Advances in Neural Information Processing Systems*, 1998, pp. 570–576.

[3] O. Maron and A. Ratan, "Multiple-instance learning for natural scene classification," in *International Conference on Machine Learning*, 1998, pp. 341–349.

[4] Q. Zhang and S. A. Goldman, "EM-DD: An improved multiple-instance learning technique," in *Advances in Neural Information Processing Systems*, 2002, pp. 1073–1080.

[5] P. Viola, J. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Advances in Neural Information Processing Systems*, 2005, pp. 1417–1424.

[6] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels," in *International Conference on Machine Learning*, 2002, pp. 179–186.

[7] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *International Journal of Computer Vision*, vol. 105, no. 3, pp. 222–245, 2013.

[8] J. Wu, Y. Yu, C. Huang, and K. Yu, "Deep multiple instance learning for image classification and auto-annotation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3460–3469.

[9] M. Ilse, J. Tomczak, and M. Welling, "Attention-based deep multiple instance learning," in *International Conference on Machine Learning*, 2018, pp. 2127–2136.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[11] Z. Shao, H. Bian, Y. Chen, Y. Wang, J. Zhang, X. Ji, et al., "TransMIL: Transformer based correlated multiple instance learning for whole slide image classification," in *Advances in Neural Information Processing Systems*, 2021, pp. 2136–2147.

[12] T. Yao, Y. Pan, Y. Li, C.-W. Ngo, and T. Mei, "Wave-ViT: Unifying wavelet and transformers for visual representation learning," in *European Conference on Computer Vision*, 2022, pp. 328–345.

[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[14] G. Campanella, M. G. Hanna, L. Geneslaw, A. Miraflor, V. W. K. Silva, K. J. Busam, et al., "Clinical-grade computational pathology using weakly supervised deep learning on whole slide

images," *Nature Medicine*, vol. 25, no. 8, pp. 1301–1309, 2019.

[15] K.-H. Yu, C. Zhang, G. J. Berry, R. B. Altman, C. Ré, D. L. Rubin, and M. Snyder, "Predicting non-small cell lung cancer prognosis by fully automated microscopic pathology image features," *Nature Communications*, vol. 7, no. 1, p. 12474, 2016.

[16] S. Graham, Q. D. Vu, S. E. A. Raza, A. Azam, Y. W. Tsang, J. T. Kwak, and N. Rajpoot, "MILD: Multi-instance learning for the detection of microsatellite instability from whole slide images," *Medical Image Analysis*, vol. 65, p. 101773, 2020.

[17] Khangle, M. (n.d.). Multiple instance learning (MIL) and its utility in whole-slide image (WSI) analyses. *Medium*. Retrieved from https://medium.com/@minhkhangle.phd/multiple-instance-learning-mil-and-its-utility-in-whole-slide-image-wsi-analyses-3acb67f5434b

[18] Chen, Junhua & Zeng, Haiyan & Zhang, Chong & Shi, Z. & Dekker, André & Wee, Leonard & Bermejo, Iñigo. (2022). Lung cancer diagnosis using deep attention-based multiple instance learning and radiomics. Medical Physics. 49. 10.1002/mp.15539.

[19] Ilse, M., Tomczak, J., & Welling, M. (2018). Attention-based deep multiple instance learning. In *International Conference on Machine Learning* (pp. 2127-2136).

[20] Campanella, G., Hanna, M. G., Geneslaw, L., Miraflor, A., Silva, V. W., Busam, K. J., ... & Fuchs, T. J. (2019). Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 25(8), 1301-1309.

[21] Lu, M. Y., Williamson, D. F., Chen, T. Y., Chen, R. J., Barbieri, M., & Mahmood, F. (2021). Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature Biomedical Engineering*, 5(6), 555-570.

[22] Moonlight, T. (n.d.). DSAGL: Dual-Stream Attention-Guided Learning for Weakly Supervised Whole-Slide Image Classification. *Themoonlight.io*. Retrieved from https://www.themoonlight.io/en/review/dsagl-dual-stream-attention-guided-learning-for-weakly-supervised-whole-slide-image-classification