

Supplementary Information for: Reinforcement Learning–Based WSI Diagnostic System for Cervical Squamous Cell Carcinoma and Precancerous Lesions

Supplementary Note 1: Pre-training Dataset Curation and Details

For this study, we constructed a large-scale, in-house dataset of cervical pathology biopsy images for model pre-training, encompassing a wide spectrum of cervical histopathological types. The data were sourced from histopathologically confirmed cervical biopsy specimens collected between 2018 and 2024, totaling **21,002 slides**. The distribution is as follows: **8,407 benign** (including inflammatory and glandular changes), **5,245 Low-grade Squamous Intraepithelial Lesion (LSIL)**, **4,203 High-grade Squamous Intraepithelial Lesion (HSIL)**, and **3,147 Squamous Cell Carcinoma (SCC)**. The dataset includes typical cases of benign lesions, LSIL, HSIL, and SCC, as well as tissue slides featuring concomitant inflammatory responses and glandular abnormalities.

All slides were stained with standard Hematoxylin & Eosin (H&E) and digitized using a high-throughput whole-slide scanner at 20× magnification. For the purpose of model pre-training and learning of salient regions, each Whole Slide Image (WSI) was tiled into non-overlapping 256×256 pixel patches. All patches underwent a manual quality control process to exclude those with excessive background or scanning artifacts, thereby ensuring the integrity and representativeness of the tissue information.

This pre-training dataset provided the subsequent CerviScan-ViT model with rich morphological diversity and structural priors of cervical tissue. By covering the multi-stage pathological continuum from normal squamous epithelium to cancerous regions, it laid a solid foundation for the model’s generalization and feature capture abilities in cervical histopathology diagnosis.

Supplementary Note 2: Mathematical Formulation and Computational Justification of SmartPatchRL

To provide a rigorous computational justification for the proposed RL-guided patch selection mechanism, we formulate the patch selection process as a **Sequential Decision-Making Process** modeled by a Markov Decision Process (MDP). Unlike heuristic selection strategies (e.g., random sampling or max-pooling) that rely on static assumptions, our method explicitly optimizes for **Information Gain** regarding the ground-truth category.

Supplementary Table S1. Performance Comparison of Different MIL Models (95% Confidence Intervals)

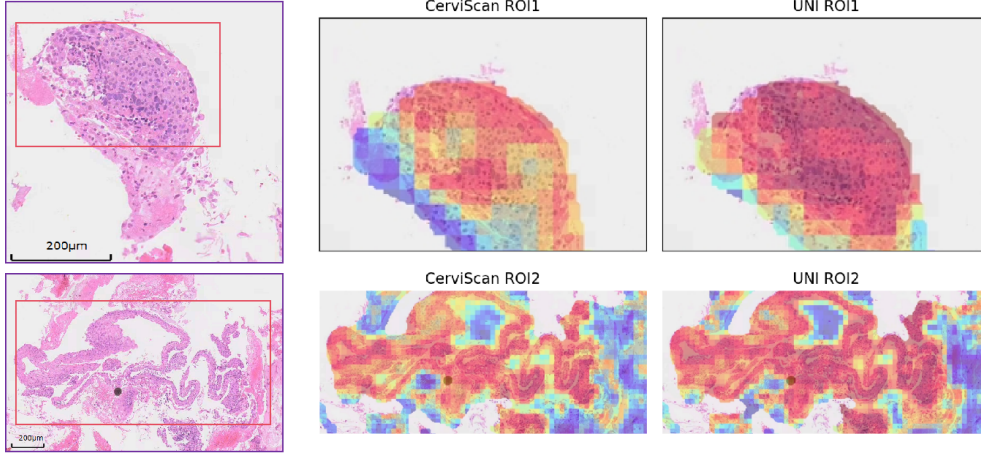
Model	Accuracy	Macro F1-score	Macro AUC
AB_MIL	(0.7213, 0.8204)	(0.7106, 0.8095)	(0.9219, 0.9380)
AC_MIL	(0.6858, 0.8029)	(0.6750, 0.7929)	(0.8930, 0.9239)
AMD_MIL	(0.7490, 0.8029)	(0.7383, 0.7930)	(0.9146, 0.9333)
CLAM_SB_MIL	(0.7725, 0.8143)	(0.7682, 0.8119)	(0.9230, 0.9438)
GATE_AB_MIL	(0.7133, 0.8027)	(0.6975, 0.7875)	(0.9185, 0.9327)
MAX_MIL	(0.6611, 0.7302)	(0.6591, 0.7268)	(0.8683, 0.9027)
MEAN_MIL	(0.5872, 0.6255)	(0.5884, 0.6291)	(0.8371, 0.8560)
TRANS_MIL	(0.7127, 0.8023)	(0.7063, 0.7982)*	(0.9084, 0.9345)*
WIKG_MIL	(0.7167, 0.7977)	(0.7032, 0.7897)	(0.9064, 0.9252)

Note: Values represent the 95% confidence interval. Ranking is determined by the upper bound of the interval. For each metric, **Bold** indicates the best performance, Underlined indicates the second-best, and * indicates the third-best result. Citations are synchronized with the main manuscript.

Supplementary Table S2. Performance Comparison Highlighting the Impact of the Feature Extractor

Feature Extractor	Aggregator	Accuracy	Macro F1-score	Macro AUC
UNI	AB_MIL	0.9025 $\pm$ 0.0109* (0.8890, 0.9159)	0.9049 $\pm$ 0.0110* (0.8913, 0.9185)	0.9841 $\pm$ 0.0030* (0.9804, 0.9878)
UNI	AC_MIL	0.8938 $\pm$ 0.0075 (0.8845, 0.9031)	0.8971 $\pm$ 0.0079 (0.8873, 0.9069)	0.9810 $\pm$ 0.0018 (0.9788, 0.9832)
UNI	AMD_MIL	0.9045 $\pm$ 0.0102 (0.8918, 0.9172)	0.9081 $\pm$ 0.0105 (0.8951, 0.9211)	0.9841 $\pm$ 0.0023* (0.9813, 0.9870)
UNI	GATE_AB_MIL	0.9013 $\pm$ 0.0067 (0.8930, 0.9095)	0.9030 $\pm$ 0.0071 (0.8941, 0.9119)	0.9836 $\pm$ 0.0027 (0.9802, 0.9870)
UNI	MAX_MIL	0.8772 $\pm$ 0.0066 (0.8690, 0.8855)	0.8794 $\pm$ 0.0073 (0.8704, 0.8884)	0.9734 $\pm$ 0.0038 (0.9687, 0.9781)
UNI	MEAN_MIL	0.8493 $\pm$ 0.0022 (0.8465, 0.8520)	0.8573 $\pm$ 0.0022 (0.8546, 0.8599)	0.9664 $\pm$ 0.0015 (0.9646, 0.9683)
UNI	TRANS_MIL	0.8923 $\pm$ 0.0087 (0.8815, 0.9032)	0.8949 $\pm$ 0.0075 (0.8856, 0.9043)	0.9821 $\pm$ 0.0050 (0.9759, 0.9883)
UNI	WIKG_MIL	0.8988 $\pm$ 0.0081 (0.8887, 0.9089)	0.9016 $\pm$ 0.0086 (0.8910, 0.9123)	0.9819 $\pm$ 0.0035 (0.9775, 0.9863)
UNI	CLAM_SB_MIL	0.9022 $\pm$ 0.0119 (0.8875, 0.9169)	0.9041 $\pm$ 0.0116 (0.8897, 0.9185)	0.9844 $\pm$ 0.0039 (0.9796, 0.9892)
CerviScan-ViT	CLAM_SB_MIL	0.9063 $\pm$ 0.0047 (0.9005, 0.9121)	0.9086 $\pm$ 0.0043 (0.9033, 0.9140)	0.9869 $\pm$ 0.0007 (0.9860, 0.9877)

Note: All baseline models use a generic UNI feature extractor. Our proposed model pairs the **CLAM_SB_MIL** aggregator with our custom **CerviScan-ViT** extractor. For the UNI baselines, Underlined indicates the best performance and * indicates the second-best. Values are presented as mean $\pm$ std, with the 95% confidence interval on the second line. Ranking is determined by the mean.



Supplementary Fig. S1. Direct comparison of heatmap interpretation on a region with significant tissue artifacts. (Right) The UNI-based model overemphasizes tissue artifacts, such as overlapping tissue and preparation-induced distortions, leading to over-grading. (Left) Our model correctly ignores these misleading cues, demonstrating superior robustness.

1. MDP Formulation

We define the interaction between the patch-selection agent and the WSI environment at step t within a finite horizon T as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{R})$:

- **State (s_t):** The state represents the agent’s internal belief state, encapsulating the diagnostic context accumulated up to step t . Let $\mathcal{B}$ be the full bag of patches available in the WSI and $\mathcal{X}_t \subset \mathcal{B}$ be the subset selected at step t . We define $z_t = \text{Agg}(\mathcal{X}_t)$ as the aggregated feature representation of the current selection. To capture the search history, the state is updated recurrently using a Gated Recurrent Unit (GRU):

$$s_t = \text{GRU}(z_t, s_{t-1}) \quad (1)$$

- **Action (a_t):** The policy network $\pi_\theta(a_t | s_t)$ outputs a selection vector (action) that determines the sampling weights or indices for the next subset of patches $\Delta\mathcal{X}_{t+1}$ to be added to the bag.
- **Reward (r_t):** The immediate feedback signal used to update the policy, designed to measure the diagnostic value of the newly selected patches.

2. Optimization Objective: Maximizing Confidence Increment

The core computational advantage of our framework lies in the specific design of the reward function. Drawing inspiration from the fine-tuning strategy in MuRCL, we define the reward r_t as the **incremental gain in classifier confidence** for the ground-truth class y :

$$r_t = P(y | s_t) - P(y | s_{t-1}) \quad (2)$$

56 where $P(y \mid s_t)$ is the softmax probability of the true class y given the current
 57 aggregated state s_t . This reward mechanism strictly incentivizes the agent to select
 58 patches that increase the model’s certainty about the correct diagnosis.

59 3. Justification via Policy Gradient

60 The objective of the RL agent is to maximize the expected cumulative reward $J(\theta)$
 61 over the episode:

$$J(\theta) = \mathbb{E}_{\pi_\theta} \left[\sum_{t=1}^T \gamma^{t-1} r_t \right] \quad (3)$$

62 Assuming a discount factor $\gamma \approx 1$ for the short horizon of patch selection steps,
 63 and substituting the reward definition from Eq. (2), the cumulative reward telescopes
 64 as follows:

$$\sum_{t=1}^T r_t = \sum_{t=1}^T (P(y \mid s_t) - P(y \mid s_{t-1})) = P(y \mid s_T) - P(y \mid s_0) \quad (4)$$

65 **Conclusion:** This derivation mathematically proves that **maximizing the**
 66 **cumulative reward is equivalent to maximizing the final classification**
 67 **probability $P(y \mid s_T)$.**

68 Unlike random selection, where the expected reward is near zero (as randomly
 69 selected patches are unlikely to systematically improve confidence), the policy gradient
 70 update:

$$\nabla_\theta J(\theta) \approx \sum_{t=1}^T \nabla_\theta \log \pi_\theta(a_t \mid s_t) \cdot A_t \quad (5)$$

71 forces the policy distribution π_θ to shift towards patch subsets that yield a positive
 72 Advantage ($A_t > 0$). This computationally guarantees that the agent converges to a
 73 policy that identifies the **most discriminative regions** (Highest Evidence), thereby
 74 justifying its superior performance over random or fixed-grid selection strategies.

75 Supplementary Note 3

Algorithm 1: Training Procedure of SmartPatchRL with PPO

Input: WSI feature bags $\mathcal{D}$, Horizon $T = 5$, PPO Clip $\epsilon = 0.2$, Epochs E

Output: Optimized Policy π_θ and Value function V_ϕ

```
1 Initialize Actor  $\pi_\theta$ , Critic  $V_\phi$ , Feature Extractor, and Classifier;
2 for epoch = 1 to  $E$  do
3   foreach WSI bag  $B \in \mathcal{D}$  do
4     Initialize state  $s_0$  (e.g., from random patch subset);
5     /* Interaction Phase: Collect Trajectory */
6     for step  $t = 1$  to  $T$  do
7       Sample action  $a_t \sim \pi_\theta(\cdot|s_t)$ ;
8       Select patch subset  $\mathcal{X}_t$  based on  $a_t$  from  $B$ ;
9       Aggregate features:  $z_t \leftarrow \text{CLAM}(\mathcal{X}_t)$ ;
10      Predict class prob:  $p_t \leftarrow \text{Classifier}(z_t)$ ;
11      Update State (GRU):  $s_t \leftarrow \text{Update}(z_t, s_{t-1})$ ;
12      /* Calculate Reward: Confidence Gain */
13      Compute Reward:  $r_t = p_t[y_{true}] - p_{t-1}[y_{true}]$ ;
14      Store transition  $(s_{t-1}, a_t, r_t, s_t, p_t)$  in memory;
15    end
16  end
17  /* Optimization Phase: PPO Update */
18  Compute Advantages  $\hat{A}_t$  using GAE (Generalized Advantage Estimation);
19  foreach optimization step do
20    Select mini-batch of transitions;
21    Compute ratio  $ratio = \frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}$ ;
22    Compute  $L^{CLIP}$  and Value Loss  $L^{VF}$ ;
23    Update  $\theta, \phi$  by maximizing  $L^{CLIP} - c_1 L^{VF} + c_2 L^S$ ;
24  end
25 end
```
