MExD: An Expert-Infused Diffusion Model for Whole-Slide Image Classification

Jianwei Zhao[1], Xin Li[2], Fan Yang[2,#], Qiang Zhai[3], Ao Luo[4], Yang Zhao[1], Hong Cheng[1], and Huazhu Fu[5]

UESTC[1], AIQ[2], SICAU[3], SWJTU[4], IHPC, A*STAR[5]

1.Introduction

- **Why WSI:**

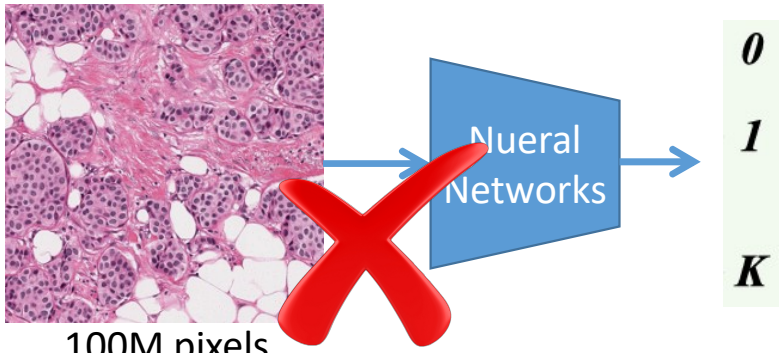
Histological Whole Slide Image (WSI) classification is essential in digital pathology, automating diagnosis and sub-typing while extracting insights from high-resolution scansto support prognosis and treatment planning

- **Features:**

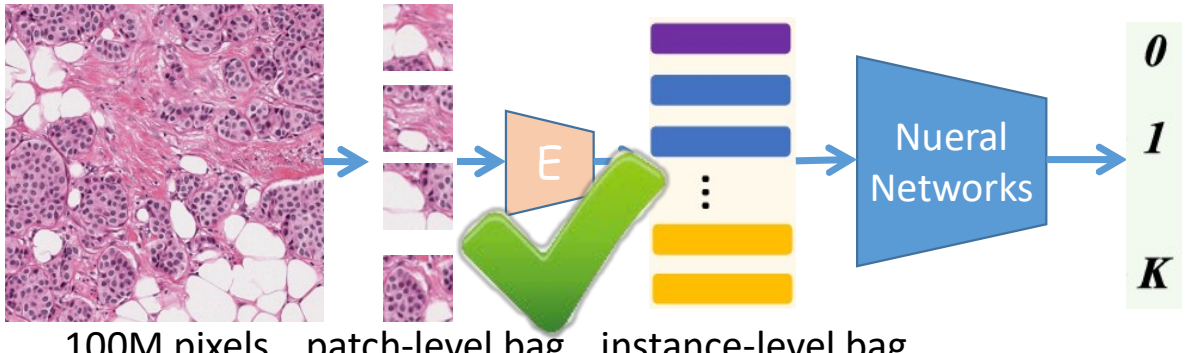
WSIs, with sizes ranging from 100 million to 10 billion pixels, are too large for conventional deep learning techniques, which were originally designed for much smaller natural or medical images.

- **Strategy:**

"decompose and aggregate" , A promising application of this strategy is Multiple Instance Learning(MIL), which treats each WSI as a bag containing smaller image patches.



Traditional solutions



Multiple Instance Learning(MIL)

1.Introduction

Motivation

A significant issue is **data imbalance:** within a bag, positive instances (e.g., those containing can-cerous cells) are often vastly outnumbered by negative ones. This imbalance can bias the model toward the majority class, leading it to overlook or misclassify important minority instances and resulting in poor sensitivity for detecting rare but clinically significant features.

The current shortcomings of **feature integration methods:** These methods are still limited by the influence of noise and irrelevant patches, as they fundamentally rely on discriminative feature integration paradigms that are inherently susceptible to such interference.

Solution

Mixture-of-Experts (MoE): which intelligently routes instances to specialized experts based on their relevance and characteristics. This dynamic routing mechanism ensures focused attention on minority classes, enhancing sensitivity to rare but clinically significant instances and reducing bias toward the majority class.

Diffusion Classifier: leverages the expert and prior knowledge extracted by Dyn-MoE through a customized diffusion process, supported by a multi-layer perceptron (MLP) network. This integration of expert driven insights with the generative capabilities of the diffusion model

1.Introduction

Overall, the contributions of this work can be summarized as:

- **A new paradigm for WSI classification.** MExD is the first to apply generative techniques for WSI classifcation,offering a novel solution to data imbalance, noise reduction, and feature integration, and laying the foundation forfuture research.

- **A novel diffusion model fused with the MoE approach.** We introduce a novel diffusion model seamlessly integrated with Mixture-of-Experts(MoE), which, unlikestandard diffusion models, incorporates experts focusedon specific instances, thereby more effectively addressing the inherent challenges of WSI analysis.

- **State-of-the-art results on widely-used benchmarks.** MExD demonstrates exceptional performance in accurately classifying WSI, setting new standards on prominent benchmarks such as Camelyonl6,TCGA-NSCLC and BRACS.

2.Method

- **Problem Reformulation：**

A WSI (X) is divided into a patch-wise *bag* **B** of *N* patches $\{x_i\}^N$, with only a bag-level label Y . For binary classification, Y = 1 indicates at least one positive instance, while Y = 0 indicates none. In multi-label tasks, Y corresponds to the specific cancer subtype, with eachWSI containing only one type.

- **MIL paradigm:**

1) Patch Embedding: Extractor $f(\cdot)$ transforms patches $\{x_i\}^N$ into embeddings $\{b_i\}^N$ forming instance-wise bag B;

2) Feature Aggregation: Aggregator $g(\cdot)$ combines embeddings;

3) WSI Classification: Classifier $c(\cdot)$ predicts the label $Y:=c(g(\{f(x_i)\}^N))$. In some methods, the classifierc$(\cdot)$ is incorporated into $g(\cdot)$.

2.Method

MEXD:

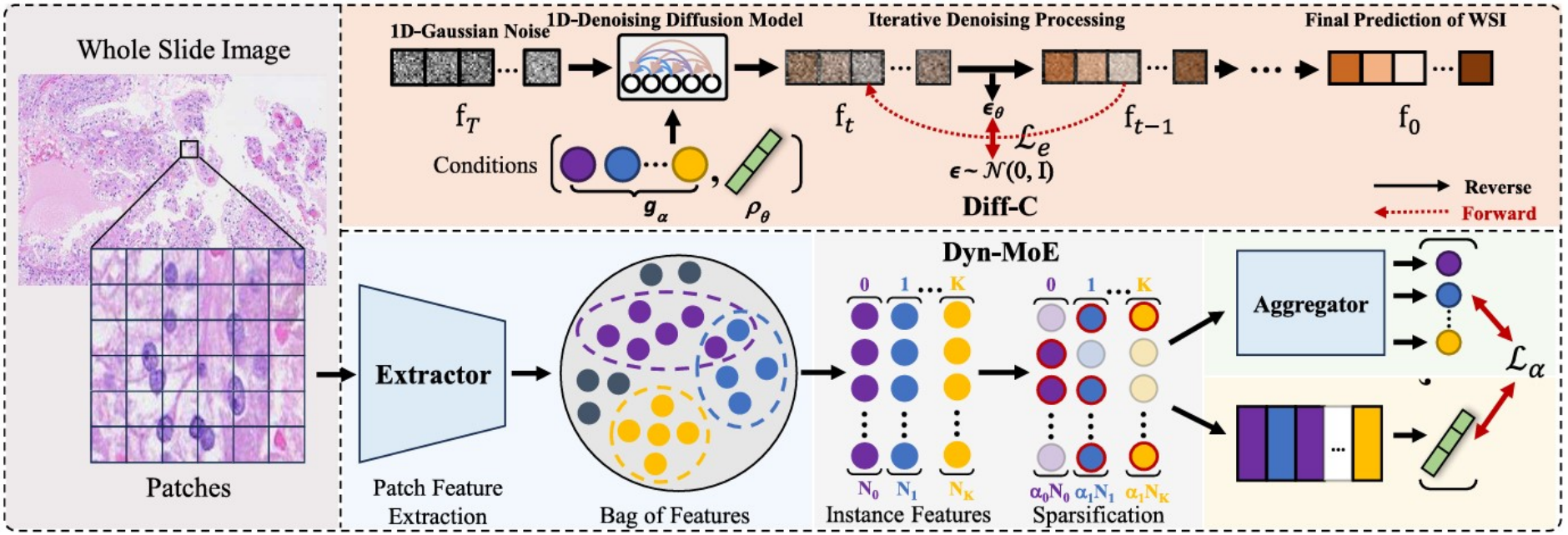


Figure 2. **Overview of MExD**. Our framework leverages a generative classification diffusion model for enhanced reliability and generalizability, where g_α and ρ_θ encode expert insights and prior knowledge, conditioning Diff-C (see Sec. 3.2 for details).

Fig. 2 presents an overview of our MExD, which integrates a Patch Feature Extractor, a Dyn-MoE Aggregator leveraging the Mixture-of-Experts mechanism, and a Diffusion Classifier (Diff-C) built on the principles of the Denoising Diffusion Probabilistic Model.

2.Method

Dyn-MOE：

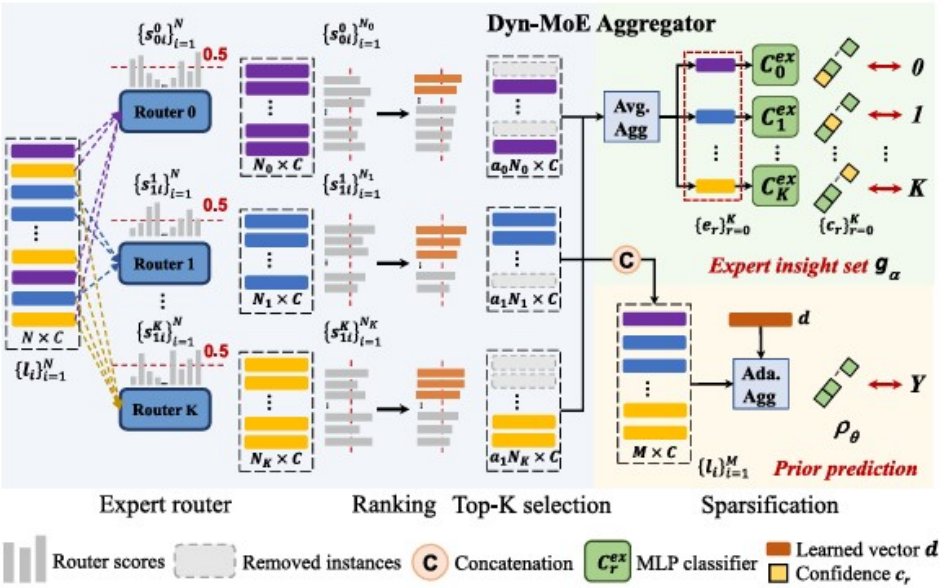


Figure 3. **Details of Dyn-MoE Aggregator**. Our Dyn-MoE includes K *positive* experts and 1 *negative* expert, producing expert insights set (g_α) and prior prediction (ρ_θ) to work alongside Diff-C. 'Avg.' and 'Ada.' indicate mean-pooling and Adapter. *Zoom in for details.*

WorkFlow:

- **Patch Feature Extractor.** Given a patch-wise *bag* $\breve{B} = \{x_i\}_{i=1}^N$, the Patch Feature Extractor, similar to conventional discriminative methods, encodes each patch to generate high-order instance features $\{b_i\}_{i=1}^N$, which are then fed into the aggregation stage.
- **Dyn-MoE Aggregator.** The Dyn-MoE Aggregator receives $\{b_i\}_{i=1}^N$ as input and dynamically samples the most representative instances using a tailored MoE mechanism, yielding a sparse set of transformed instances $\{l_i\}_{i=1}^M$, where $M < \frac{N}{2}$. It then learns the concatenation $[\{l_i\}_{i=1}^M, d]$ to produce prior prediction ρ_θ, which captures the relationship between the WSI (X) and the bag-level label (Y), with d as a learnable class embedding. Meanwhile, each expert sub-branch generates a class-centric latent feature e and confidence c, forming the expert insights set $g_\alpha = \{(e_i, c_i)\}_{i=0}^K$.
- **Diffusion Classifier.** Our Diff-C utilizes a modified forward diffusion process, incorporating prior prediction ρ_θ as a predicted conditional expectation, along with a multilayer perceptron (MLP) serving as a noise prediction network conditioned on the expert insights set g_α, to directly predict the final result $\hat{Y}$.

2.Method

Forward Diffusion

(a) *Forward Diffusion with ρ_θ.* Forward diffusion generates a series of noised samples $\boldsymbol{y}_t$ at timestamps $\boldsymbol{t}$ from the initial signal $\boldsymbol{y}_0$, following a Markov process as shown in the first line of Eq. 5. As $\boldsymbol{t}$ increases from 1 to T, $\boldsymbol{y}_t$ can also be derived directly from $\boldsymbol{y}_0$ using the second line of Eq. 5:

$$\begin{aligned} q(\boldsymbol{y}_t|\boldsymbol{y}_{t-1}) &= \mathcal{N}(\boldsymbol{y}_t; \tilde{\beta}_t \boldsymbol{y}_{t-1},\ \beta_t \mathbf{I}), \\ q(\boldsymbol{y}_t|\boldsymbol{y}_0) &= \mathcal{N}(\boldsymbol{y}_t; \sqrt{\overline{\alpha}_t}\boldsymbol{y}_0,\ (1-\overline{\alpha}_t)\mathbf{I}). \end{aligned} \tag{5}$$

Here, β_t is the variance of the Gaussian noise at each step $\boldsymbol{t}$, with $\tilde{\beta}_t = \sqrt{1-\beta_t}$, $\overline{\alpha}_t = \prod_{j=1}^{t} \alpha_j$, and $\alpha_j = 1-\beta_j$. As $\overline{\alpha}_T \to 0$ at the maximum timestamp T, the endpoint of FD, $\boldsymbol{y}_T$ becomes pure Gaussian noise. Inspired by CARD [16], we assume the one-hot encoded label $\mathbf{f}_0$ with $\boldsymbol{\rho}_\theta$ as the predicted conditional expectation, infusing it into FD to specify conditional distributions in Eq. 6 (top), allowing closed-form sampling at any timestamp $\boldsymbol{t}$ in Eq. 6 (bottom):

$$\begin{aligned} q(\mathbf{f}_t|\mathbf{f}_{t-1}, \boldsymbol{\rho}_\theta) &= \mathcal{N}(\mathbf{f}_t; \tilde{\beta}_t \mathbf{f}_{t-1} + (1-\tilde{\beta}_t)\boldsymbol{\rho}_\theta,\ \beta_t \mathbf{I}), \\ q(\mathbf{f}_t|\mathbf{f}_0, \boldsymbol{\rho}_\theta) &= \mathcal{N}(\mathbf{f}_t; \sqrt{\overline{\alpha}_t}\mathbf{f}_0 + (1-\sqrt{\overline{\alpha}_t})\boldsymbol{\rho}_\theta, (1-\overline{\alpha}_t)\mathbf{I}). \end{aligned} \tag{6}$$

Intuitively, in MExD, the endpoint of FD is the prior prediction $\boldsymbol{\rho}_\theta$.

Reverse Denoising :

(b) *Reverse Denoising with g_α.* Diff-C uses a denoising network $\mathcal{D}$, implemented as a three-layer MLP, to predict the added noise $\boldsymbol{\epsilon}_t$. It receives expert insights set $\boldsymbol{g}_\alpha$ and an initial noise input $\mathbf{f}_T \sim \mathcal{N}(\boldsymbol{\rho}_\theta, \mathbf{I})$. First, we compute a weighted average of all $\boldsymbol{e}$ values in $\boldsymbol{g}_\alpha$ using their corresponding confidences $\boldsymbol{c}$, which is fed into an encoder with three linear layers to produce the conditioned latent $\boldsymbol{Z} \in \mathbb{R}^{1\times C}$. The denoising network iteratively estimates the intermediate noise ϵ_θ, refining $\mathbf{f}_t$ to $\mathbf{f}_{t-1}$ following the FD posterior from CARD [16], until the reconstructed output $\mathbf{f}_0^{'}$ is obtained:

$$q(\mathbf{f}_{t-1}|\mathbf{f}_t, \mathbf{f}_0, \boldsymbol{\rho}_\theta) = \mathcal{N}(\mathbf{f}_{t-1}; \boldsymbol{\mu}(\mathbf{f}_t, \mathbf{f}_0, \boldsymbol{\rho}_\theta), \hat{\beta}_t \mathbf{I}). \tag{7}$$

Based on Eq. 7, the entire process can be mathematically defined as follows:

$$\begin{aligned} \boldsymbol{Z} &= \sum_{r=0}^{k} (\boldsymbol{c}_r \cdot \boldsymbol{e}_r), \quad \boldsymbol{\epsilon}_\theta = \mathcal{D}(\boldsymbol{Z}, \mathbf{f}_t, \boldsymbol{\rho}_\theta, \boldsymbol{t}), \\ \hat{\mathbf{f}}_0 &= \frac{1}{\sqrt{\overline{\alpha}_t}}(\mathbf{f}_t - (1-\sqrt{\overline{\alpha}_t})\boldsymbol{\rho}_\theta - (1-\sqrt{\overline{\alpha}_t})\boldsymbol{\epsilon}_\theta), \\ \mathbf{f}_{t-1} &= \gamma_0 \hat{\mathbf{f}}_0 + \gamma_1 \mathbf{f}_t + \gamma_2 \boldsymbol{\rho}_\theta + \gamma_3 \boldsymbol{z}. \end{aligned} \tag{8}$$

Here, $\boldsymbol{z}$ is a standard Gaussian noise. γ_0, γ_1, γ_2 and γ_3 are associated with parameters α and β[1]. In the context of the reverse denoising process, as formalized in Eq. 8, the Diff-C progressively refines the noisy prediction through an iterative sequence, *i.e.*, $\mathbf{f}_T \to \mathbf{f}_{T-\Delta} \to \cdots \mathbf{f}_0$.

3.Quantitative Results

Table 1. **Quantitative Comparison of Our Results and State-of-the-Art Methods** on CAMELYON16, TCGA-NSCLC, and BRACS datasets. All models are retrained under their recommended settings for fairness, except for PAMIL [54], whose results are directly cited. The best performance is in **bold**, and the second best is underlined.

Method	Camelyon16			TCGA-NSCLC			BRACS		
	F1-score	ACC	AUC	F1-score	ACC	AUC	F1-score	ACC	AUC
CTransPath pre-trained with SRCL									
ABMIL [19]	89.81±0.69	90.70±0.63	92.41±0.78	89.67±0.81	89.71±0.80	95.87±0.57	62.54±1.16	68.29±0.86	80.44±0.80
DSMIL [23]	88.84±0.73	89.73±0.74	92.05±1.17	90.65±0.73	90.67±0.72	96.40±0.53	69.73±1.82	72.82±1.06	86.70±0.43
CLAM-SB [30]	90.47±0.62	91.32±0.65	93.55±0.86	94.17±1.07	94.37±1.06	96.90±0.74	68.37±1.57	69.65±1.29	84.04±0.61
CLAM-MB [30]	91.20±0.75	91.94±0.69	93.63±0.38	93.04±0.71	93.05±0.73	97.13±0.47	67.61±2.14	69.18±1.53	80.78±1.30
TransMIL [38]	93.69±0.56	94.19±0.44	95.03±0.67	90.94±0.88	90.95±0.89	95.89±0.23	70.17±1.80	72.00±1.29	85.18±0.56
DTFD-MIL [50]	94.80±0.39	95.16±0.29	96.82±0.58	88.31±0.31	88.38±0.26	94.98±0.76	68.98±3.02	72.00±1.75	83.57±1.18
MHIM-MIL [42]	94.67±0.68	94.99±0.76	96.14±0.55	91.34±0.81	91.68±0.49	96.73±0.65	72.41±1.63	73.26±0.96	84.79±0.82
MambaMIL [47]	93.98±0.71	94.42±0.65	96.06±0.78	94.23±0.72	94.48±0.64	97.16±0.40	68.37±0.84	70.59±0.83	83.39±1.01
ACMIL [52]	93.80±0.32	93.99±0.38	94.85±0.72	94.08±1.01	94.10±0.98	96.80±0.76	72.88±1.07	74.35±0.52	85.70±0.73
IBMIL [28]	95.02±0.65	95.35±0.63	96.41±1.13	93.52±0.32	93.62±0.26	97.45±0.53	70.93±0.95	72.94±0.83	85.84±0.71
WiKG [26]	90.62±1.20	91.28±0.97	92.73±1.82	93.23±0.53	93.24±0.52	95.95±1.24	69.00±0.60	71.06±0.64	84.19±0.59
PAMIL* [54]	96.10±0.60	94.70±0.90	97.70±2.60	‡	‡	‡	‡	‡	‡
MExD	**97.29±0.41**	**97.48±0.37**	**98.87±0.43**	**96.51±0.29**	**96.53±0.23**	**98.13±0.17**	**75.17±0.42**	**76.13±0.58**	**88.08±0.26**
ViT pre-trained with MoCo V3									
ABMIL [19]	85.04±1.06	87.02±0.74	84.29±0.98	88.41±0.32	88.57±0.34	92.49±0.46	52.39±0.47	67.77±0.65	78.43±0.30
DSMIL [23]	78.82±0.65	81.20±0.97	82.64±1.43	90.23±0.85	90.09±0.78	95.83±0.25	54.01±0.92	63.77±0.99	80.07±0.83
CLAM-SB [30]	90.32±0.95	91.16±0.88	95.15±0.71	94.38±0.53	94.38±0.52	96.93±0.23	69.67±1.54	71.76±1.66	83.22±0.42
CLAM-MB [30]	87.78±1.34	88.99±1.15	96.64±1.18	93.70±1.04	93.72±1.01	97.35± 0.50	71.46±2.79	73.65±1.78	79.82±2.14
TransMIL [38]	93.33±0.60	93.80±0.63	95.05±0.44	93.51±0.71	93.62±0.72	96.40±0.41	74.04±0.31	74.82±0.64	87.15±0.39
DTFD-MIL [50]	89.98±1.30	90.89±1.16	93.05±0.27	89.67±1.05	89.72±1.04	94.60±0.73	49.93±2.48	62.35±1.44	83.22±0.63
MHIM-MIL [42]	93.24±1.40	93.70±0.77	97.20±0.60	90.48±1.50	92.17±0.81	94.95±0.53	62.76±1.31	71.28±1.26	86.20±0.69
MambaMIL [47]	93.73±0.86	94.11±0.88	97.28±0.51	94.68±0.35	94.76±0.33	97.76±0.36	61.63±2.48	69.88±1.05	85.62±0.84
ACMIL [52]	86.69±0.72	87.75±0.65	89.44±0.42	95.09±0.79	95.33±0.52	97.74±0.20	71.25±0.88	74.59±1.05	84.64±0.49
IBMIL [28]	92.68±1.06	93.33±0.88	94.37±0.71	94.23±0.43	94.29±0.34	97.46±0.39	74.74±0.61	75.53±0.98	86.57±0.27
WiKG [26]	86.28±0.96	87.60±0.55	90.06±1.45	92.73±0.87	92.76±0.85	97.52±0.44	63.99±2.15	66.83±1.54	82.52±0.88
MExD	**96.49±0.38**	**96.74±0.34**	**98.00±0.16**	**95.87±0.25**	**95.87±0.24**	**98.27±0.32**	**80.10±0.48**	**81.01±0.45**	**88.61±0.42**

1.MExD achieves leading performance across all metrics and datasets, setting new benchmarks.

Table 2. **Quantitative Uncertainty Assessment** using PAvPU (α-value=0.05) [34], conducted via hypothesis testing with CTransPath [44] as $\boldsymbol{f}_{\text{PFE}}$.

Method	Camelyon16		TCGA-NSCLC		BRACS	
	ACC	PAvPU	ACC	PAvPU	ACC	PAvPU
ACMIL[52]	94.57	93.80	94.29	93.81	74.12	76.47
IBMIL [28]	94.57	93.02	93.81	94.29	74.12	75.29
TransMIL [38]	94.57	94.57	91.90	91.90	72.94	72.94
MambaMIL [47]	95.35	95.35	94.76	94.76	71.76	71.76
MExD	**97.67**	**98.45**	**96.67**	**97.62**	**76.47**	**81.18**

2.MExD shows full certainty in correct predictions and uncertainty in some incorrect ones

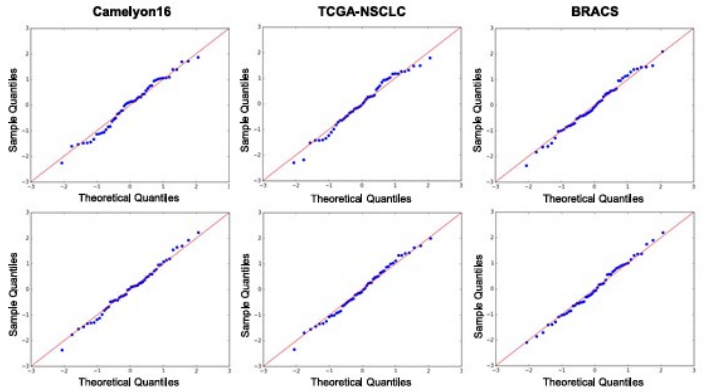


Figure 5. **Normality Assumption Assessment:** Q-Q plots showing the probability difference between the top two predicted classes within a *bag*, with two *bags* selected from each benchmark for visualization.

3.Qualitative Interpretability

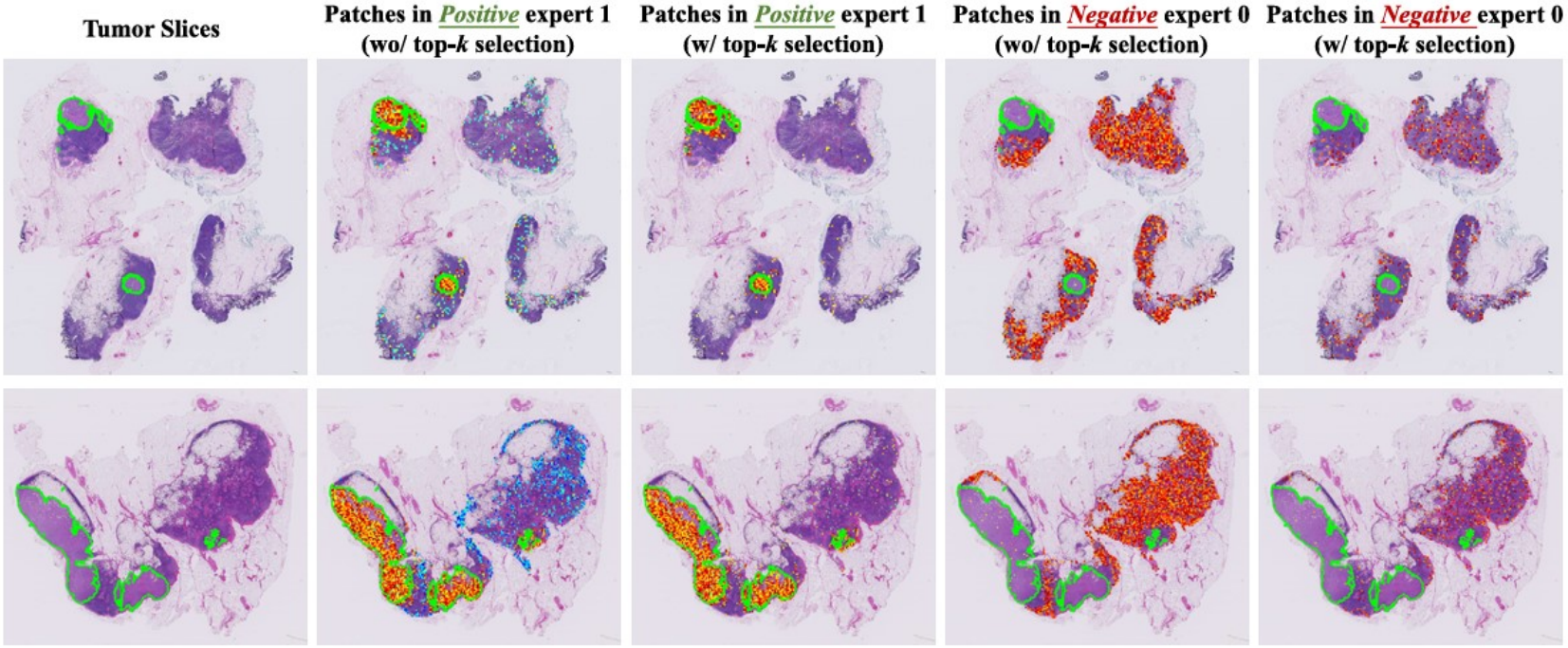
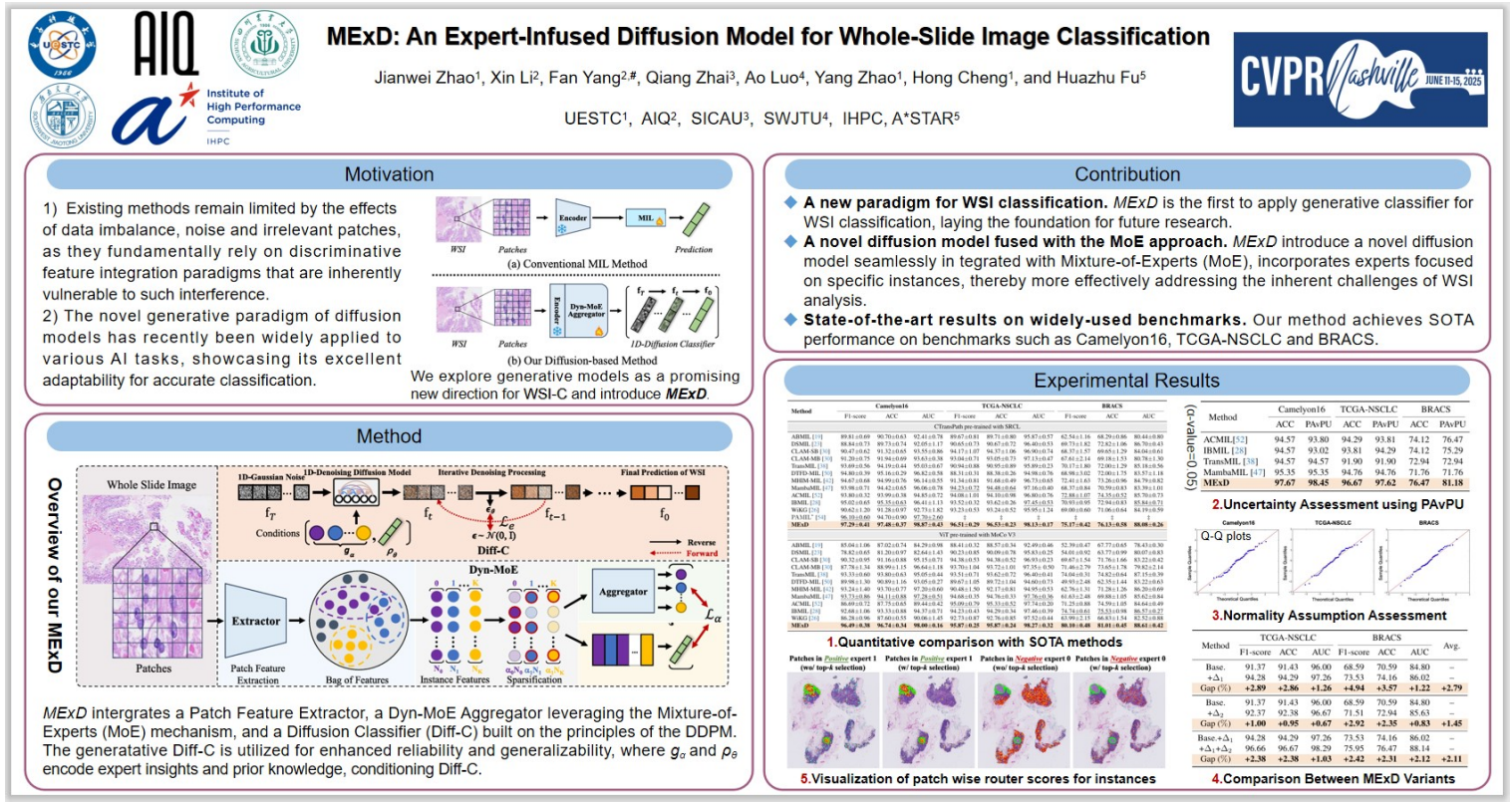


Figure 4. **Distribution Visualization** of patch-wise router scores for *positive* instances, where each score corresponds to a selected patch. In column 2, expert 1 effectively identifies *positive* patches, then refines and retains the most representative ones (column 3) through score-based selection. Concurrently, expert 0 focuses on refining *negative* instances. The green edges indicate cancerous region.

In column 2, router scores from our positive expert1 are mapped onto the WSI, and patches without scores are excluded. Expert 1 identifes positive patches effectively, despite some negative interference, achieving instance sparsity. Notably, true positive patches receive higher scores, while false positives are scored lower. Column 3 demonstrates score-based refnement, where only higher-scoring patches are retained.

4.Conclusion

In this work, we introduce MExD, a novel WSI classifcation framework that combines a conditional generative approach with Mixture-of-Experts (MoE). MExD includes a Dyn-MoE aggregator that dynamically selects relevant instances, addressing data imbalance and improving representation. Additionally, its generative Diff-C iteratively denoises Gaussian noise using expert insights and prior predictions, resulting in stable outcomes. This blend of MoE and generative classification offers a promising new approach to WSI classifcation with broader applications incomputer vision.



Poster

ExHall D Poster #474

Sat 14 Jun 5 p.m. — 7 p.m. CDT

Thanks for your watching!