

Probabilistic smooth attention for deep multiple instance learning in medical imaging

Francisco M. Castro-Macías^{a,c}, Pablo Morales-Álvarez^{b,c}, Yunan Wu^{d,e}, Rafael Molina^{a,c}, Aggelos K. Katsaggelos^{d,e}

^a*Department of Computer Science and AI, University of Granada, Spain*

^b*Department of Statistics and Operations Research, University of Granada, Spain*

^c*Research Centre for Information and Communication Technologies (CITIC), University of Granada, Spain*

^d*Department of Electrical and Computer Engineering, Northwestern University, USA*

^e*Center for Computational Imaging and Signal Analytics in Medicine, Northwestern University, USA*

Abstract

The Multiple Instance Learning (MIL) paradigm is attracting plenty of attention in medical imaging classification, where labeled data is scarce. MIL methods cast medical images as bags of instances (e.g. patches in whole slide images, or slices in CT scans), and only bag labels are required for training. Deep MIL approaches have obtained promising results by aggregating instance-level representations via an attention mechanism to compute the bag-level prediction. These methods typically capture both local interactions among adjacent instances and global, long-range dependencies through various mechanisms. However, they treat attention values deterministically, potentially overlooking uncertainty in the contribution of individual instances. In this work we propose a novel probabilistic framework that estimates a probability distribution over the attention values, and accounts for both global and local interactions. In a comprehensive evaluation involving eleven state-of-the-art baselines and three medical datasets, we show that our approach achieves top predictive performance in different metrics. Moreover, the probabilistic treatment of the attention provides uncertainty maps that are interpretable in terms of illness localization.

Keywords: Multiple instance learning, probabilistic machine learning, Bayesian methods, Whole Slide Images, CT scans

1. Introduction

Multiple Instance Learning (MIL) is a popular machine learning framework in which the model is trained on sets of instances, called bags, rather than individual instances [1]. Unlike the traditional supervised learning framework, which needs the label of each instance, in MIL only the label of the bag is required. This approach is particularly beneficial in scenarios where labelling many individual instances is impractical or prohibitively expensive. This is especially relevant in the context of medical imaging, where annotations are extremely costly [2]. Fig. 1 shows two illustrative examples of MIL problems within this context: (i) cancer detection from Whole Slide Images (WSIs), where the WSI represents the bag, and the patches are the instances; and (ii) intracranial hemorrhage detection from Computerized Tomographic (CT) scans, where the full scan represents the bag and the slices at different heights are the instances.

Several methods have been proposed for learning in the MIL scenario, with Deep Learning (DL) approaches achieving remarkable results [2]. A seminal work in the area was the one by Ilse et al. [3], which proposes the attention-based MIL (ABMIL) method. ABMIL features an attention mechanism with the ability to focus on the most relevant instances. The attention assigned to each instance can serve as a proxy to find instances with a positive label, i.e., those showing evidence of the disease or injury. With this in mind, the attention mechanism has been improved in various ways [4, 5, 6]. However, these methods have a major limitation: they are unable to capture interactions between instances because the attention mechanism treats them independently. Ultimately, this negatively affects the discriminative performance of these approaches. Based on the existing literature, two types of interactions among instances have been considered: global and local.

Global interactions are those that may exist between any pair of instances in a bag. Shao et al. [7] showed that they can contain important information and proposed the TransMIL model, which includes a Transformer encoder to capture them. Note that the self-attention mechanism used in Transformers is a natural

choice to model these interactions since it computes a similarity score between each pair of instances. For this reason, different variations of this mechanism were included to boost the performance of existing MIL models [8, 9, 10]. *Local interactions* are those that exist between neighboring instances. A natural way of modeling them is to treat each bag as a graph, where the instances are the nodes and the edges between them represent these local interactions. This is the approach followed by recent works, which have exploited them in combination with global interactions [9, 11, 12, 13] and alone using Graph Neural Networks [14, 15]. Note that an important difference between global and local interactions is that global interactions are *learned* from the data, while local interactions are known beforehand and encoded into the model.

Recently, we proposed an interesting way of leveraging local interactions, called Smooth Attention (SA) [16]. It is based on the idea that, in CT scans, a slice is usually adjacent to slices with the same label, see Fig. 1b. Since the attention values act as a proxy to estimate these labels, they should inherit this property. To enforce this spatial constraint, SA proposes a novel regularization term based on the Dirichlet energy [17], which is used to train the ABMIL model. It obtained very promising results in the intracranial hemorrhage (ICH) detection task, outperforming the vanilla ABMIL and other related approaches. However, SA does not account for global correlations, and it estimates attention values deterministically.

The deterministic nature of SA and the other previously mentioned methods prevents them from expressing uncertainty in their predictions. However, the ability to model uncertainty is a crucial requirement in medical diagnosis. To address this limitation, we propose the Probabilistic Smooth Attention (ProbSA) framework – a probabilistic generalization of SA – which introduces the following key contributions:

1. ProbSA provides a general probabilistic formulation. When a deterministic procedure is used for inference, we recover SA as a particular case. When Bayesian inference is leveraged instead, we obtain a new method

that estimates attention values through a full probability distribution. As we will see, this yields enhanced predictive performance and allows for uncertainty estimation in the predictions.

2. ProbSA handles both local and global interactions. We explain how the local interactions in [16] can be combined with global interactions to yield a new method superior to the current state-of-the-art (SOTA) approaches, which usually leverage both types of interactions.
3. We provide a comprehensive evaluation as required for archival work. ProbSA is evaluated using three different datasets (two more than [16]) covering two different medical imaging problems: cancer detection in WSIs and hemorrhage detection in CT scans. Also, ProbSA is compared against ten SOTA methods in MIL (six more than those used in [16]).

The rest of the paper is organized as follows. Sec. 2 reviews the foundations of our work (ABMIL and SA). Sec. 3 presents the novel ProbSA framework, discussing the probabilistic model and inference as well as the combination with global interactions. Sec. 4 discusses the empirical evaluation of the model, including dataset description, experimental setup, ablation study to understand ProbSA, and comparison against SOTA methods. Sec. 5 provides the main conclusions and limitations.

2. Background

Sec. 2.1 describes the MIL problem and reviews the attention-based MIL model (ABMIL) [3], which is at the basis of the proposed approach. Sec. 2.2 describes our deterministic approach to introduce smoothness in the attention values, previously published as Smooth Attention (SA).

2.1. Attention-based Multiple Instance Learning

In MIL, the training set consists of pairs of the form (\mathbf{X}, Y) , where $\mathbf{X} \in \mathbb{R}^{N \times P}$ is a bag of instances and $Y \in \{0, 1\}$ is the bag label. Each bag is composed

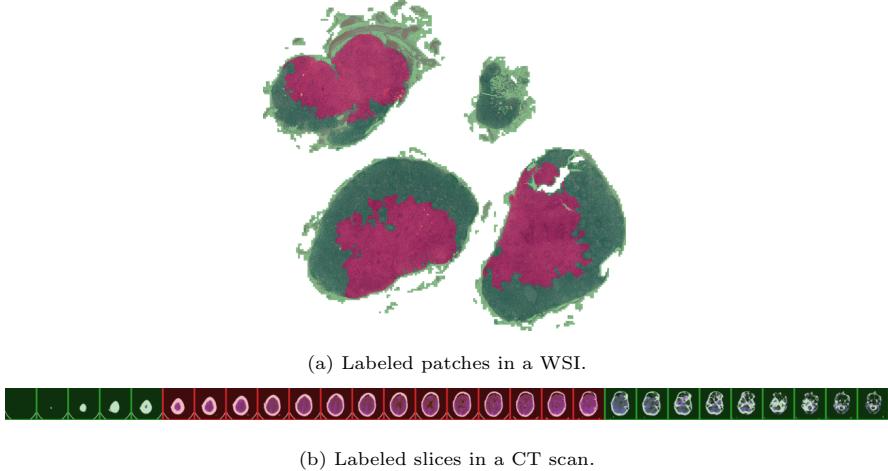


Figure 1: Two modalities of MIL medical images: (a) a whole slide image (WSI) for tumour detection and (b) a CT scan for hemorrhage detection. The red color indicates malignant/hemorrhage patches/slices, respectively. During training, instance labels are not known, and only a global bag-level label is available. Note that these labels show spatial dependencies: a patch/slice is usually surrounded by patches/slices with the same label.

of a variable number of instances, i.e., $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T$, where $\mathbf{x}_n \in \mathbb{R}^P$ are the instances. Each instance \mathbf{x}_n has an *unobserved* label $y_n \in \{0, 1\}$ which is not available during training. It is assumed that $Y = \max \{y_1, \dots, y_N\}$, that is, a bag \mathbf{X} is considered positive if and only if there is at least one positive instance in the bag. The goal is to learn a function that predicts the label of previously unseen bags.

Attention-based MIL (ABMIL) was proposed by Ilse et al. [3] to learn in this scenario. This method maps the input bag \mathbf{X} to a bag representation $\mathbf{z} \in \mathbb{R}^D$, which is later used to predict the bag label. To compute this representation they use the attention-based pooling, which assigns an *attention value* f_n to each instance \mathbf{x}_n . This value indicates its importance within the bag and can be used as a proxy to estimate the instance label y_n . Formally,

$$\mathbf{z} = \mathbf{H}^\top \text{Softmax}(\mathbf{f}), \quad (1)$$

where $\mathbf{H} = [h(\mathbf{x}_1), \dots, h(\mathbf{x}_N)]^\top$, $h: \mathbb{R}^P \rightarrow \mathbb{R}^D$ is a neural network applied to each instance, $\mathbf{f} = [f_1, \dots, f_N]^\top$, and $f_n = \mathbf{w}^\top \tanh(\mathbf{W}\mathbf{h}_n)$, with $\mathbf{W} \in \mathbb{R}^{D_f \times D}$

and $\mathbf{w} \in \mathbb{R}^{D_f}$ being trainable weights.

Note that in ABMIL both the intermediate embeddings \mathbf{h}_n and the attention values \mathbf{f} are obtained by applying a transformation *independently* to each instance. As a result, they will be the same regardless of what other instances are in the bag or what structure the bag has. In other words, the dependencies between the instances are neglected.

2.2. Smooth Attention (SA): deterministic attention smoothing

In order to take into account the dependencies between the instances during training, Smooth Attention (SA) [16] proposes to leverage the following property of medical images: if an instance shows evidence of a lesion, neighboring instances likely contain it too, see Fig. 1. Therefore, for the attention maps to estimate the instance labels accurately, they should also exhibit this *smoothing* property: the attention value assigned to an instance should be similar to the values assigned to surrounding instances.

SA restricts the solution space by discarding solutions with highly variable attention maps. This is achieved through the *Dirichlet energy*, a well-known functional that measures the variability of a function defined on a graph [17]. For a function $\mathbf{f} \in \mathbb{R}^N$ defined on a graph with adjacency matrix $\mathbf{A} = [A_{ij}] \in \mathbb{R}^{N \times N}$, the Dirichlet energy of \mathbf{f} is given by,

$$\mathcal{E}_D(\mathbf{f}, \mathbf{A}) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N A_{ij} (f_i - f_j)^2 = \mathbf{f}^\top \mathbf{L} \mathbf{f}, \quad (2)$$

where \mathbf{L} is the graph Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, and $\mathbf{D} \in \mathbb{R}^{N \times N}$ is the degree matrix, $\mathbf{D} = \text{Diag}(D_1, \dots, D_N)$, $D_n = \sum_i A_{ni}$. SA treats the attention values as a function defined on the bag graph, which is defined by $A_{ij} > 0$ if instances \mathbf{x}_i and \mathbf{x}_j are neighbors, and $A_{ij} = 0$ otherwise.

The Dirichlet energy is used to define a regularization term that penalizes solutions with highly variable attention maps. Formally, given a dataset of bags $\{\mathbf{X}_1, \dots, \mathbf{X}_B\}$, with corresponding bag labels $\{Y_1, \dots, Y_B\}$ and adjacency matrices $\{\mathbf{A}_1, \dots, \mathbf{A}_B\}$, SA uses the architecture proposed in ABMIL and seeks

to minimize the following objective,

$$\sum_{b=1}^B \{-\log p(Y_b | \mathbf{X}_b) + \lambda \mathcal{E}_D(\mathbf{f}_b, \mathbf{A}_b)\}, \quad (3)$$

where $p(Y_b | \mathbf{X}_b) = \text{Bernoulli}\left(Y_b | \psi\left(\mathbf{H}_b^\top \text{Softmax}(\mathbf{f}_b)\right)\right)$, $\lambda > 0$ is an hyper-parameter, $\psi: \mathbb{R}^D \rightarrow [0, 1]$ is the bag classifier, \mathbf{H}_b and \mathbf{f}_b are as in Eq. (1).

SA achieved very promising results in the task of hemorrhage detection, and here we extend it in two ways. First, since its architecture is based on ABMIL, it does not include global interactions, which has proven successful in previous works. Second, like the rest of current SOTA deep MIL methods, it is a deterministic approach. By estimating a probability distribution over the attention values, we will achieve enhanced discriminative performance as well as interpretable uncertainty estimations. In the following section, we propose the Probabilistic Smooth Attention (ProbSA) framework, a generalization of SA that addresses these two aspects.

3. Probabilistic Smooth Attention (ProbSA)

In this section, we describe the novel Probabilistic Smooth Attention framework. Sec. 3.1 describes the Bayesian formulation of the model. In Sec. 3.2 we describe how to include global interactions in the architecture of the proposed methodology. Finally, Sec. 3.3 provides an overview of the different model variants proposed in this work.

3.1. Model formulation

We assume we are given a training set of bags $\mathcal{X} = \{\mathbf{X}_1, \dots, \mathbf{X}_B\}$, with corresponding bag labels $\mathcal{Y} = \{Y_1, \dots, Y_B\}$ and adjacency matrices $\mathcal{A} = \{\mathbf{A}_1, \dots, \mathbf{A}_B\}$. Recall that $\mathbf{X}_b \in \mathbb{R}^{N_b \times D}$, $Y_b \in \{0, 1\}$, and $\mathbf{A}_b \in \mathbb{R}^{N_b \times N_b}$. Since we will be dealing with bags of different sizes, in the following we denote $\mathbb{R}^{\mathbb{N} \times D} = \cup_{m \in \mathbb{N}} \mathbb{R}^{m \times D}$, $\mathbb{R}^{\mathbb{N}} = \cup_{m \in \mathbb{N}} \mathbb{R}^m$, and $(0, +\infty)^{\mathbb{N}} = \cup_{m \in \mathbb{N}} (0, +\infty)^m$.

Probabilistic model and inference. We model the attention values of each bag \mathbf{X}_b as an unobserved latent variable \mathbf{f}_b . The smoothing property is encoded

as a prior distribution given by

$$p(\mathbf{f}_b \mid \mathbf{A}_b) \propto \exp(-\mathcal{E}_D(\mathbf{f}_b, \mathbf{A}_b)). \quad (4)$$

This corresponds to the conditional and simultaneous autoregressive models in the statistics literature [18]. Following ABMIL [3], the attention values are used to compute the bag label likelihood,

$$p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b) = \text{Bernoulli}\left(Y_b \mid \psi(\mathbf{H}_b^\top \text{Softmax}(\mathbf{f}_b))\right), \quad (5)$$

where $\mathbf{H}_b = H(\mathbf{X}_b)$, $H: \mathbb{R}^{N \times P} \rightarrow \mathbb{R}^{N \times D}$ is a bag transformation, and $\psi: \mathbb{R}^D \rightarrow [0, 1]$ is the bag classifier. Note that in the original ABMIL and in SA the bag transformation H is implemented by applying a neural network independently to every instance. We will explain how to include global interactions in H in Sec. 3.2. Finally, the joint probabilistic model is obtained assuming independence across bags,

$$p(\mathcal{Y}, \mathcal{F} \mid \mathcal{X}, \mathcal{A}) = \prod_{b=1}^B p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b) p(\mathbf{f}_b \mid \mathbf{A}_b), \quad (6)$$

where $\mathcal{F} = \{\mathbf{f}_1, \dots, \mathbf{f}_B\}$. Making predictions in this model requires computing the posterior of the bag label given the observations. More precisely, given a new bag \mathbf{X}^* , the corresponding bag label Y^* is obtained from,

$$p(Y^* \mid \mathbf{X}^*, \mathcal{Y}) = \int p(Y^* \mid \mathbf{X}^*, \mathbf{f}^*) p(\mathbf{f}^* \mid \mathbf{X}^*, \mathcal{Y}) d\mathbf{f}^*. \quad (7)$$

However, computing the posterior $p(\mathbf{f} \mid \mathbf{X}, \mathcal{Y})$ in closed form is not possible. To address this, we follow the Variational Inference (VI) [19] approach: we approximate it using a variational distribution $q(\mathbf{f} \mid \mathbf{X}, \mathcal{Y})$, which we write as $q(\mathbf{f} \mid \mathbf{X})$ in what follows. Using the variational distribution, we can make predictions using

$$p(Y^* \mid \mathbf{X}^*, \mathcal{Y}) \approx \int p(Y^* \mid \mathbf{X}^*, \mathbf{f}^*) q(\mathbf{f}^* \mid \mathbf{X}^*) d\mathbf{f}^* = \quad (8)$$

$$= \mathbb{E}_{q(\mathbf{f}^* \mid \mathbf{X}^*)} [p(Y^* \mid \mathbf{X}^*, \mathbf{f}^*)]. \quad (9)$$

The optimal choice of $q(\mathbf{f} \mid \mathbf{X})$ is obtained by minimizing the Kullback-Leibler (KL) divergence between the true posterior and the variational distribution [19].

This is equivalent to maximizing the following Evidence Lower BOund (ELBO),

$$\text{ELBO} = \sum_{b=1}^B \mathbb{E}_{q(\mathbf{f}_b | \mathbf{X}_b)} \left[\log \frac{p(Y_b | \mathbf{X}_b, \mathbf{f}_b) p(\mathbf{f}_b | \mathbf{A}_b)}{q(\mathbf{f}_b | \mathbf{X}_b)} \right]. \quad (10)$$

Unfortunately, it is not possible to obtain the optimal variational distribution in closed form. Instead, we proceed as in [20]: we restrict $q(\mathbf{f} | \mathbf{X})$ to belong to some parameterized family of distributions and adjust its parameters by maximizing the ELBO in Eq. (10). In the following, we consider two choices for this family: as a Gaussian and as a Dirac delta.

Modelling $q(\mathbf{f} | \mathbf{X})$ as a Gaussian. In this case, we parameterize the variational distribution as a multivariate Gaussian distribution with a diagonal covariance matrix,

$$q(\mathbf{f} | \mathbf{X}) = \mathcal{N}(\mathbf{f} | \mu(\mathbf{X}), \Sigma(\mathbf{X})), \quad \Sigma(\mathbf{X}) = \text{Diag}(\sigma(\mathbf{X})), \quad (11)$$

where $\mu: \mathbb{R}^{N \times D} \rightarrow \mathbb{R}^N$ and $\sigma: \mathbb{R}^{N \times D} \rightarrow (0, +\infty)^N$ are implemented using neural networks. With this choice, the ELBO in Eq. (10) can be written as

$$\text{ELBO} = \sum_{b=1}^B \left\{ \mathbb{E}_{q(\mathbf{f}_b | \mathbf{X}_b)} [\log p(Y_b | \mathbf{X}_b, \mathbf{f}_b)] - \text{KL}[q(\mathbf{f}_b | \mathbf{X}_b), p(\mathbf{f}_b | \mathbf{A}_b)] \right\}. \quad (12)$$

Since sampling from this distribution is very efficient, we can approximate the expectations in the first term and Eq. (9) – Eq. (10) using Monte Carlo sampling and the reparameterization trick [20]. Moreover, the KL divergence in the second term admits the following closed-form expression,

$$\text{KL}[q(\mathbf{f}_b | \mathbf{X}_b), p(\mathbf{f}_b | \mathbf{A}_b)] = \mathcal{E}_D(\mu(\mathbf{X}_b), \mathbf{A}_b) + \text{Tr}(\mathbf{L}_b \Sigma(\mathbf{X}_b)) + \quad (13)$$

$$- \frac{1}{2} \left(\log \left(\frac{|\Sigma(\mathbf{X}_b)|}{2|\mathbf{L}_b|} \right) - N_b \right), \quad (14)$$

which can be computed efficiently. Note that the KL divergence contains the Dirichlet energy used by SA plus a new term that regularizes the covariance matrix. Informally speaking, if we make the covariance matrix very small, $\Sigma(\cdot) \rightarrow 0$, the KL divergence becomes the regularization term used by SA, see Eq. (3). This can be formalized by modeling $q(\mathbf{f} | \mathbf{X})$ as a Dirac delta, as follows.

Recovering SA: $q(\mathbf{f} \mid \mathbf{X})$ as a Dirac delta. We define the variational distribution as a *deterministic distribution*, i.e., a distribution with zero variance. To do so, we resort to the notion of Dirac delta [21]. However, since it is not a proper function we cannot use it to define our variational density. Fortunately, there exists mathematical machinery to use this concept rigorously [21, 22], see Appendix A for a precise justification of the following statements. We define

$$q(\mathbf{f} \mid \mathbf{X}) = \delta(\mathbf{f} - \mu(\mathbf{X})) = \begin{cases} \infty & \text{if } \mathbf{f} = \mu(\mathbf{X}), \\ 0 & \text{otherwise,} \end{cases} \quad (15)$$

where $\delta(\cdot)$ is the Dirac delta and $\mu: \mathbb{R}^{N \times D} \rightarrow \mathbb{R}^N$ is implemented using a neural network. Informally speaking, $q(\mathbf{f} \mid \mathbf{X})$ assigns all the probability to one point which is determined by the output of μ . The ELBO becomes

$$\text{ELBO} = \sum_{b=1}^B \{\log p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b = \mu(\mathbf{X}_b)) - \mathcal{E}_D(\mu(\mathbf{X}_b), \mathbf{A}_b)\}, \quad (16)$$

which is the negative of the objective used by SA with $\lambda = 1$, recall Eq. (3). Therefore, the probabilistic model ProbSA presented in this section has our previous SA [16] as a particular case.

The loss objective. Whatever choice we make for $q(\mathbf{f} \mid \mathbf{X})$ (as a Gaussian or as a Dirac delta), we must maximize the corresponding ELBO with respect to the parameters of the neural network. In both cases, the negative of the ELBO leads to the following minimization objective,

$$\mathcal{L}_{\text{ELBO}} = -\text{ELBO} = \mathcal{L}_{\text{LL}} + \mathcal{L}_{\text{KL}}, \quad (17)$$

where \mathcal{L}_{LL} is the negative log-likelihood and \mathcal{L}_{KL} is a KL regularization term, see Table 1. Previous works have introduced a balancing hyperparameter $\lambda \in [0, 1]$ to control the strength of the regularization and alleviate KL collapse [23]. Following this idea, we propose to minimize the following objective,

$$\mathcal{L} = \mathcal{L}_{\text{LL}} + \lambda \mathcal{L}_{\text{KL}}. \quad (18)$$

Note that by setting $\lambda = 1$ we recover the negative of the ELBO, and by setting $\lambda = 0$ we drop the KL regularization. To avoid the KL vanishing problem, some

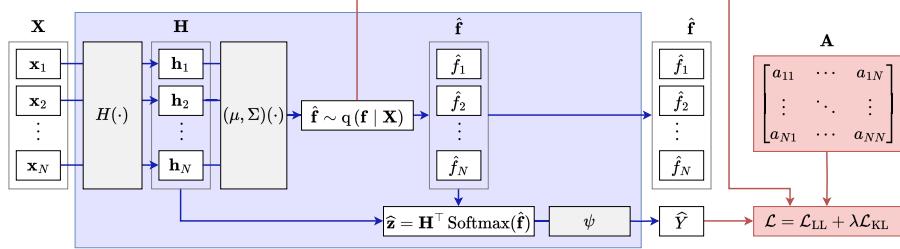


Figure 2: Diagram of the proposed ProbSA. In the forward pass (blue), a sample $\hat{\mathbf{f}}$ is drawn from the variational attention distribution $q(\mathbf{f} \mid \mathbf{X})$ and then used to compute the bag representation $\hat{\mathbf{z}}$ and the predicted bag label \hat{Y} . In the loss computation (red), spatial interactions are incorporated via the bag adjacency matrix into de Kullback-Leibler divergence term.

works have proposed to vary the value of λ during training, following different schedules [24]. We will use the cyclical annealing schedule proposed by Fu et al. [24], and analyze its impact on the performance in Sec. 4.2.

Making predictions. To predict the bag label Y^* of a new bag \mathbf{X}^* , we must approximate the integral in Eq. 9. The variational posterior choices we have presented facilitate efficient sampling. Thus, we draw S samples $\{\mathbf{f}_1^*, \dots, \mathbf{f}_S^*\}$ from $q(\mathbf{f}^* \mid \mathbf{X}^*)$ and compute

$$p(Y^* \mid \mathbf{X}^*, \mathcal{Y}) \approx \frac{1}{S} \sum_{s=1}^S p(Y^* \mid \mathbf{X}^*, \mathbf{f}_s^*). \quad (19)$$

This process is illustrated in Fig. 2. For a given input bag, one or more samples are drawn from the variational attention distribution and then used to compute the predicted bag label according to Eq. 19.

3.2. Incorporating global interactions

Both SA and the proposed ProbSA take into account local interactions through a regularization term in the training objective, see Eq. (18). However, previous works in deep MIL have demonstrated the benefits of including both global and local interactions [9, 11]. Note that local interactions provide information about the environment of each instance but do not account for long-range dependencies. For example, detecting some types of tumors requires

finding certain structures in distant regions within the WSI [7, 9]. Next, we explain how we can include global interactions in the proposed ProbSA.

As evidenced in Fig. 2, one of the main advantages of ProbSA is that it is *architecture agnostic*. This means that we can choose any architecture to implement the bag transformation H , the bag classifier ψ , or the attention distribution networks (μ, Σ) , and still use the same training objective. Therefore, global interactions can be included by implementing H as a Transformer encoder,

$$H(\mathbf{X}) = \text{TransformerEnc}(\mathbf{X}). \quad (20)$$

At the core of the Transformer encoder module is the self-attention mechanism, which processes the input by calculating similarity scores between all pairs of instances in the bag. These scores determine how much attention each instance should pay to others, allowing the model to capture (global) contextual relationships effectively. Each layer of the Transformer encoder leverages multiple heads in the self-attention mechanism, along with skip-connections and pre-layer normalization [25]. More details can be found in Appendix B.

3.3. Summary of the proposed models

In the previous subsections, we introduced key design choices within the ProbSA framework that define different model variants. Specifically, there are two options for the variational posterior $q(\mathbf{f} | \mathbf{X})$ and two for the bag transformation H . These choices can be naturally combined, resulting in four distinct model variants.

Table 1 summarizes these variants and the fundamental differences between them. ABMIL and T-ABMIL stand for the two choices for the bag transformation H . Similarly, $\Sigma = 0$ and $\Sigma = \text{Diag}$ refer to the two choices for the variational posterior $q(\mathbf{f} | \mathbf{X})$. Note that we write $\Sigma = 0$ for the Dirac delta variational posterior since we recover it by making the variance infinitely small in the Gaussian, $\Sigma(\cdot) \rightarrow 0$. The variants with $\Sigma = 0$ can be regarded as an extension of the deterministic method proposed in our previous work [16]. They are deterministic because they learn a degenerate distribution for the attention

Notation	$H(\mathbf{X})$	$q(\mathbf{f} \mid \mathbf{X})$	\mathcal{L}_{LL}	\mathcal{L}_{KL}
ABMIL+ProbSA $\Sigma = 0$	$[h(\mathbf{x}_1), \dots, h(\mathbf{x}_N)]^\top$ h is a MLP	$\delta(\mathbf{f} - \mu(\mathbf{X}))$	$-\sum_b \log p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b = \mu(\mathbf{X}_b))$	$\sum_b \mathcal{E}_D(\mu(\mathbf{X}_b), \mathbf{A}_b)$
ABMIL+ProbSA $\Sigma = \text{Diag}$	$[h(\mathbf{x}_1), \dots, h(\mathbf{x}_N)]^\top$ h is a MLP	$\mathcal{N}(\mathbf{f} \mid \mu(\mathbf{X}), \Sigma(\mathbf{X}))$	$-\sum_b \mathbb{E}_{q(\mathbf{f}_b \mid \mathbf{X}_b)} [\log p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b)]$	$\sum_b \text{KL} [q(\mathbf{f}_b \mid \mathbf{X}_b), p(\mathbf{f}_b \mid \mathbf{A}_b)]$ Eq.(13)-(14)
T-ABMIL+ProbSA $\Sigma = 0$	TransformerEnc(\mathbf{X})	$\delta(\mathbf{f} - \mu(\mathbf{X}))$	$-\sum_b \log p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b = \mu(\mathbf{X}_b))$	$\sum_b \mathcal{E}_D(\mu(\mathbf{X}_b), \mathbf{A}_b)$
T-ABMIL+ProbSA $\Sigma = \text{Diag}$	TransformerEnc(\mathbf{X})	$\mathcal{N}(\mathbf{f} \mid \mu(\mathbf{X}), \Sigma(\mathbf{X}))$	$-\sum_b \mathbb{E}_{q(\mathbf{f}_b \mid \mathbf{X}_b)} [\log p(Y_b \mid \mathbf{X}_b, \mathbf{f}_b)]$	$\sum_b \text{KL} [q(\mathbf{f}_b \mid \mathbf{X}_b), p(\mathbf{f}_b \mid \mathbf{A}_b)]$ Eq.(13)-(14)

Table 1: Different variants of the proposed methodology. Each variant is determined by the bag transformation H and the variational posterior $q(\mathbf{f} \mid \mathbf{X})$. The latter yields different forms of the objective terms \mathcal{L}_{LL} and \mathcal{L}_{KL} . See Fig. 2 for a graphical illustration.

values. In contrast, the variants with $\Sigma = \text{Diag}$ learn a Gaussian distribution, from which we can sample and study its moments, see Sec. 4.4. In the following section, we conduct extensive experimentation to evaluate these four variants.

4. Experiments

In this section, we evaluate the proposed method on three real medical image classification problems, comparing it against ten SOTA MIL methods. Our code will be made publicly available upon the acceptance of the paper at the following Github repository: <https://github.com/Franblueeee/ProbSA-MIL>. In Sec. 4.1 we describe the experimental framework. In Sec. 4.2 we perform an ablation study to better understand the proposed method. In Sec. 4.3 we compare our method against ten SOTA methods in MIL. Finally, Sec. 4.4 shows how the probabilistic treatment can help in by pointing out regions of the attention maps where the method may not be confident enough.

4.1. Experimental framework

Datasets. We evaluate the proposed method on three medical MIL datasets: RSNA [26], PANDA [27], and CAMELYON16 [28]. In RSNA the goal is to detect acute intracranial hemorrhage from CT scans. There are 1150 CT scans, each of them having from 24 to 57 slices. They are labeled as having hemorrhage (at least one slice shows evidence of hemorrhage) or not hemorrhage (no slice shows evidence of hemorrhage). In PANDA the goal is to detect prostate cancer

from microscopy scans of prostate biopsy samples. It is composed of 10616 WSIs at $10\times$ magnification. In CAMELYON16 the goal is to detect breast cancer metastasis. It consists of 400 WSIs at $20\times$ magnification. In both PANDA and CAMELYON16, we extract patches of size 512×512 using the method proposed by Lu et al. [4]. Each WSI is labeled as tumorous (at least one patch shows evidence of tumor) and non-tumorous (no patch shows evidence of tumor). For RSNA and CAMELYON16, we use the standard train/test partition. For PANDA, since the labels for the test set have not been released, we use the train/test split proposed by Silva-Rodriguez et al. [29]. We split the initial train data into five different train/validation splits. Every model is trained on each of these splits and then evaluated on the test set. We report the average performance on this test set.

Metrics. Medical imaging datasets are typically imbalanced, containing more negative examples (those that do not show evidence of disease or lesion) than positive examples (those that show evidence of disease or lesion). For this reason, we analyze the performance of each method using the area under the ROC curve (AUROC) and the F1 score. We also report the average rank: we sort the different methods according to their performance on each tuple (metric, dataset), and report the average position.

Feature extraction. Neither the proposed method nor the methods we compare with can be trained end-to-end alongside a feature extractor. This limitation arises because, for the datasets under consideration, almost no single bag can fit within the memory of commercially available GPUs. For this reason, we extract features from each instance using a pre-trained model. To choose this model, we note that previous works have pointed out the importance of using large-scale domain-aligned feature extractors [30, 31], with several improvements in MIL [7, 6, 9]. For pathology data (such as PANDA and CAMELYON16), the mentioned studies provide the weights of these models. However, we have not been able to find them for CT scans. For this reason, for the RSNA dataset, we use a ResNet50 pre-trained on Imagenet. For PANDA and CAMELYON16, we use a ResNet50 pre-trained using the Barlow Twins self-supervised learning

method on a huge dataset of WSIs patches [30], whose weights can be found online¹. Both choices transform the input instance into a vector of $P = 2048$ components. Finally, note that we use the same features for every MIL method, which ensures a fair comparison.

MIL methods and architectures. We compare the proposed ProbSA with SOTA MIL methods. To ensure a fair comparison, we divide them into two groups, depending on whether or not they take into account global interactions through the self-attention mechanism. In the first group, we include those methods that do not account for these interactions: the proposed ABMIL+ProbSA, ABMIL [3], CLAM [4], DSMIL [6], PatchGCN [15], and DTFD-MIL [5]. In the second group we include those methods that do account for these interactions: the proposed T-ABMIL+ProbSA, TransMIL [7], SETMIL [12], GTP [11], IIBMIL [32], CAMIL [9], and VMIL [10]. Note that the proposed ProbSA outperforms the related [13], which uses the same datasets and metrics.

For every method, we use the original implementation, which is publicly available on their GitHub repositories. We modify the number of layers and their dimensions to make the comparison under the same parameter budget. Every method first transforms the input instances using one fully connected layer with 512 units ($D = 512$). The different variants of the attention pooling used by ABMIL, CLAM, DSMIL, PatchGCN, and DTFD-MIL are implemented with an inner dimension of $D_f = 128$. The different transformer encoders used by TransMIL, SETMIL, GTP, IIBMIL, and CAMIL have 2 transformer layers; key, query, and value dimensions of 512, and 8 attention heads. The bag-embedding classifier is implemented using one fully connected layer. The attention pooling used by ABMIL+ProbSA and T-ABMIL+ProbSA is the same as in ABMIL. The transformer encoder in T-ABMIL+ProbSA uses Pytorch’s implementation of dot product attention². The bag adjacency matrices (see Eq. (2)) are constructed at the beginning of training and remain static through-

¹Weights available [here](#).

²Pytorch’s implementation of dot product attention is available [here](#).

out the process—that is, they are not learned or updated during optimization. Following the approach of [9], we compute edge weights based on the inverse of the distance between corresponding instances in the feature space. This ensures that, if the feature space is appropriately structured, the weight between a positive and a negative instance will be close to zero, which suppresses attention smoothing across boundaries.

Training setup. To ensure fair and reproducible conditions, we trained each method under the same setup. Since the authors of IIBMIL do not share their training code, we implemented the training procedure following the description in the original manuscript. The number of epochs was set to 100. We used the Adam optimizer with the default Pytorch configuration. The base learning rate was set to 10^{-4} . We adopted a slow training start using Pytorch’s `LinearLR` scheduler with `start_factor=0.1` and `total_iters=10`. During training, we monitored the bag AUROC in the validation set and kept the weights that obtained the best results in terms of bag AUROC. In RSNA and PANDA, the batch size was set to 32. In CAMELYON16, it was set to 4 for non-transformer methods, and to 1 for transformer-based methods. For SETMIL, however, we had to set it to 1 in PANDA and CAMELYON16 due to its high GPU memory requirements. In RSNA and PANDA, we weighted the loss function to account for the imbalance between positive and negative bags. All experiments were performed on an NVIDIA GeForce RTX 3090.

4.2. Ablation study: analyzing the novel ProbSA

The aim of this subsection is to study the effect of three key components of our method: the baseline method on top of which it is implemented (ABMIL vs T-ABMIL), the hyperparameter λ , and the variational posterior $q(\mathbf{f} \mid \mathbf{X})$.

To analyze this, we ran the four variants described in Table 1 with $\lambda \in \{0, 0.1, 0.5, 1.0, \text{cyclical}\}$. Here, $\lambda = 0$ corresponds to the baseline models ABMIL and T-ABMIL, on which we build. Choosing $\lambda = 0.1$ removes most of the influence of the KL term, while setting $\lambda = 1.0$ corresponds to using the actual ELBO. When $\lambda = \text{cyclical}$, our method adopts the cyclical schedule, mentioned

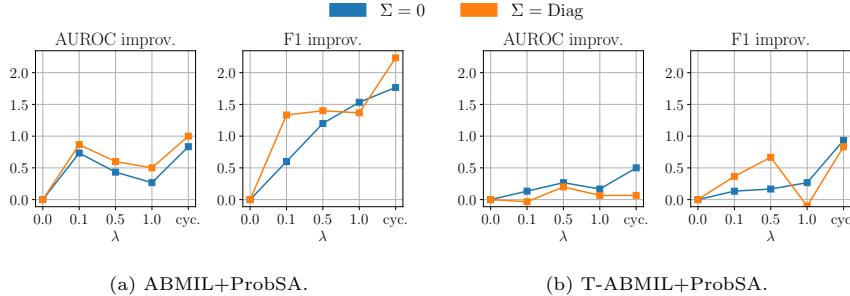


Figure 3: Average improvement over the baseline ($\lambda = 0$) for different values of the balancing hyperparameter. For most values of this hyperparameter, the improvement introduced by ProbSA is positive, regardless of the posterior choice $\Sigma = 0$ or $\Sigma = \text{Diag}$. As theoretically expected, the simpler model ABMIL benefits from a richer posterior ($\Sigma = \text{Diag}$) more than T-ABMIL does.

above. Namely, the training process is divided into $M = 5$ cycles of equal number of training steps L (L is obtained as the total number of training steps divided by M). In each cycle, the hyperparameter starts at $\lambda = 0$ and increases linearly for $\lfloor 0.8L \rfloor$ training steps until it reaches $\lambda = 1$, and stays like that for the rest of the cycle.

The results are shown in Fig. 3, where we have plotted the average improvement over the baselines ($\lambda = 0$) for different values of λ . The breakdown of these results for each dataset is in Table C.5. We analyze them in the following.

Improvement over the baselines. First, we study whether the proposed ProbSA ($\lambda > 0$) improves the results with respect to the baselines on top of which it is implemented ($\lambda = 0$). In Fig. 3, most values of λ correspond to points above zero, indicating that the proposed ProbSA improves performance compared to the baseline approaches. Notably, the improvement is negative for only two λ values in the T-ABMIL variant, with a value very close to zero. This suggests the proposed method does not significantly degrade performance in these cases. We also note that the improvement is greater for the ABMIL architecture than for T-ABMIL. This is reasonable, as simpler architectures tend to benefit more from a probabilistic formulation. When looking at Table C.5 the baselines ($\lambda = 0$) obtain the worst rank in the first three variants. The last

variant (T-ABMIL+ProbSA with $\Sigma = \text{Diag}$) does not outperform the baseline in RSNA and PANDA, but does so in CAMELYON16 by a wide margin (98.486 vs 97.673 and 94.657 vs 91.826).

Optimal value of λ . Next, we analyze the best choice for the balancing hyperparameter. We observe that $\lambda = \text{cyclical}$ provides the best performance on average: it achieves the highest improvement in Fig. 3 and the best rank in Table C.5. Note that the optimal value of λ is highly dependent on the dataset and architecture. Table C.5 suggests that for ABMIL+ProbSA higher values of λ yield better results on RSNA, while lower values are preferred for PANDA. This suggests that the cyclical scheduler will deliver a good performance in most cases, although it can be outperformed by conducting a grid search for the given dataset. This is consistent with what was observed in [24]: the cyclical schedule allows the progressive learning of more meaningful attention values. For this reason, we report the results corresponding to $\lambda = \text{cyclical}$ in the next section.

Choice of $q(\mathbf{f} \mid \mathbf{X})$, i.e., $\Sigma = 0$ vs $\Sigma = \text{Diag}$. Finally, we compare the two options considered for the variational posterior: the Gaussian ($\Sigma = \text{Diag}$) and the Dirac delta ($\Sigma = 0$). From Fig. 3 it is clear that the Gaussian provides a wider improvement when paired with the ABMIL architecture. For T-ABMIL, one could argue that the Dirac delta is a better choice in terms of AUROC, but it is not as clear for the F1 metric. However, the Gaussian posterior has an advantage over the Dirac delta: it naturally produces instance uncertainty maps, as will be discussed in Section 4.4.

4.3. Comparison against existing methods

The goal of this subsection is to compare the proposed ProbSA with SOTA deep MIL methods. As we described in Sec. 4.1, we categorize these methods into two families to ensure a fair comparison in terms of their architectural capabilities. The first group comprises methods that do not model global interactions within their architecture. This group includes our ABMIL+ProbSA. The second group consists of methods that incorporate global interactions, typically leveraging the self-attention mechanism. Our T-ABMIL+ProbSA falls

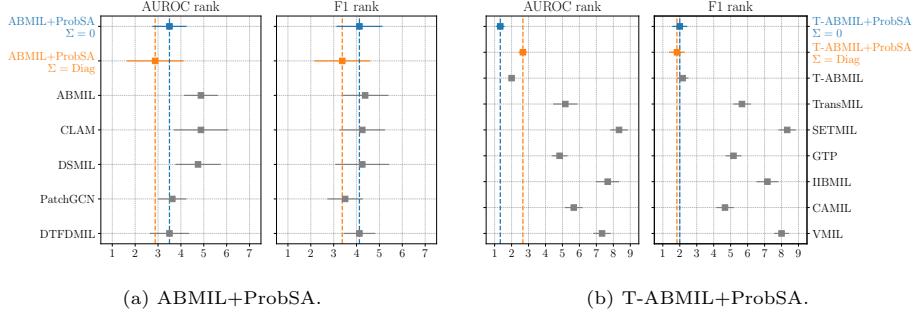


Figure 4: Average AUROC and F1 rank for each family of methods. Note that lower rank indicates better performance. The proposed ProbSA attains the best rank in each classification metric. The Gaussian variant ($\Sigma = \text{Diag}$) places first in three out of four (metric, family) pairs.

Model	RSNA		PANDA		CAMELYON16		
	AUROC (\uparrow)	F1 (\uparrow)	AUROC (\uparrow)	F1 (\uparrow)	AUROC (\uparrow)	F1 (\uparrow)	Rank (\downarrow)
ABMIL+ProbSA($\Sigma = 0$)	<u>90.189</u> _{0.482}	<u>83.168</u> _{1.259}	97.896 _{0.090}	94.973 _{0.140}	97.633 _{0.618}	91.307 _{1.053}	4.000 _{2.098}
ABMIL+ProbSA($\Sigma = \text{Diag}$)	90.231 _{0.401}	83.230 _{1.719}	98.094 _{0.069}	95.274 _{0.236}	97.885 _{1.259}	92.351 _{2.385}	1.833 _{1.329}
ABMIL	87.924 _{1.179}	78.317 _{2.531}	97.843 _{0.201}	95.060 _{0.435}	97.454 _{1.074}	90.756 _{2.483}	5.667 _{0.816}
CLAM	82.764 _{6.162}	52.941 _{32.008}	97.915 _{0.191}	95.169 _{0.176}	96.985 _{1.561}	<u>91.656</u> _{1.103}	5.333 _{1.862}
DSMIL	87.539 _{1.973}	76.164 _{5.715}	98.043 _{0.135}	95.398 _{0.269}	93.301 _{2.042}	85.801 _{3.320}	5.167 _{2.317}
PatchGCN	89.598 _{1.036}	79.748 _{2.472}	98.103 _{0.238}	<u>95.365</u> _{0.255}	97.469 _{0.873}	91.457 _{2.430}	<u>2.667</u> _{1.033}
DTFD-MIL	88.529 _{0.360}	79.348 _{1.034}	98.096 _{0.192}	95.361 _{0.298}	<u>97.821</u> _{0.532}	90.762 _{1.988}	3.333 _{1.211}

Table 2: Classification results (mean and standard deviation from five independent runs) for methods that do not model global interactions. The best is in bold and the second-best is underlined. (\downarrow)/(\uparrow) means lower/higher is better.

into this category.

Methods without global interactions. Table 2 shows the results in terms of AUROC and F1 for methods that do not account for global interactions. The average rank for each metric is shown in Fig. 4a. The proposed ProbSA with a Gaussian variational posterior ($\Sigma = \text{Diag}$) achieves the highest rank and the best result in four out of six (dataset, metric) pairs. We observe that PatchGCN achieves the best performance in PANDA and the second-best rank overall. This can be attributed to the fact that its architecture is the only one that incorporates local interactions through the graph convolutional layer.

Methods with global interactions. Table 3 shows the results in terms of

Model	RSNA		PANDA		CAMELYON16		
	AUROC (↑)	F1 (↑)	AUROC (↑)	F1 (↑)	AUROC (↑)	F1 (↑)	Rank (↓)
T-ABMIL+ProbSA($\Sigma = 0$)	91.781 _{1.200}	84.591 _{2.535}	97.974 _{0.156}	95.213 _{0.072}	98.418 _{1.104}	<u>94.220</u> _{2.168}	1.833 _{0.983}
T-ABMIL+ProbSA($\Sigma = \text{Diag}$)	90.814 _{1.683}	83.683 _{3.669}	<u>97.988</u> _{0.117}	95.339 _{0.249}	98.133 _{0.828}	94.732 _{3.071}	2.167 _{0.983}
T-ABMIL	<u>91.083</u> _{0.978}	<u>84.083</u> _{1.962}	98.014 _{0.226}	<u>95.289</u> _{0.322}	<u>98.209</u> _{0.695}	92.967 _{3.638}	<u>2.000</u> _{0.632}
TransMIL	90.271 _{0.534}	81.258 _{2.541}	96.921 _{0.239}	93.918 _{0.337}	97.811 _{2.136}	91.773 _{2.276}	5.500 _{1.225}
SETMIL	61.588 _{0.811}	12.061 _{16.515}	86.974 _{1.203}	<u>79.236</u> _{1.305}	75.340 _{0.900}	64.151 _{2.001}	8.333 _{1.033}
GTP	90.296 _{1.464}	82.324 _{4.364}	97.768 _{0.199}	94.590 _{0.694}	95.102 _{1.034}	90.096 _{1.045}	4.833 _{0.983}
IIBMIL	86.877 _{1.152}	68.290 _{6.579}	97.417 _{0.064}	94.578 _{0.182}	45.342 _{5.937}	10.971 _{24.533}	7.500 _{1.225}
CAMIL	88.686 _{2.245}	80.409 _{2.846}	97.489 _{0.111}	94.676 _{0.314}	96.811 _{1.510}	91.949 _{2.328}	5.167 _{1.169}
VMIL	89.560 _{1.153}	69.622 _{1.964}	93.763 _{0.357}	83.921 _{0.281}	50.000 _{0.000}	00.000 _{0.000}	7.667 _{1.033}

Table 3: Classification results (mean and standard deviation from five independent runs) for methods that model global interactions. The best is in bold and the second-best is underlined. (↓)/(↑) means lower/higher is better.

AUROC and F1 for methods that do not account for global interactions. The average rank for each metric is shown in Fig. 4b. The two variants of the proposed ProbSA are in the top 3. The Dirac delta variant ($\Sigma = 0$) yields the best AUROC rank, while the Gaussian variant ($\Sigma = \text{Diag}$) yields the best F1 rank, see Fig. 4b. Interestingly, the Gaussian variant performs best in PANDA and is very competitive in CAMELYON16, where it obtains the highest F1, but falls short in RSNA. We observe that IIBMIL and VMIL perform notably worse on the CAMELYON16 dataset, despite achieving competitive results on RSNA and PANDA. This discrepancy can be attributed to the challenging nature of CAMELYON16. In this dataset, bags are significantly larger – containing, on average, approximately five times more instances per bag than PANDA – and the proportion of positive instances is very low. Since both IIBMIL and VMIL rely on an instance-level classifier that requires the identification of positive instances during training, the difficulty in locating positive instances in CAMELYON16 likely hampers the training process, leading to inaccurate predictions.

Performance in RSNA. Finally, we observe that all methods perform significantly worse on the RSNA dataset compared to the other two. This discrepancy can be attributed to the feature extractor used for RSNA, which was not pre-trained on data from this domain, resulting in a feature space with reduced representational capacity.

4.4. Exploring instance predictions

We conclude the experimental section by exploring the localization capabilities of each method. Recall that deep MIL methods usually assign a scalar to each instance that reflects its importance within the bag. For simplicity, we will refer to these scalars as *attention values*, although they can be obtained in different ways (e.g., using GradCam as in GTP [11]). The proposed ProbSA, instead of assigning an attention value to each instance, outputs a probability distribution, denoted as $q(\mathbf{f} | \mathbf{X})$. In this section, we show that the first and second-order moments of the Gaussian variational posterior $\mathcal{N}(\mu(\mathbf{X}), \Sigma(\mathbf{X}))$ provide important information for illness localization.

To show this, we analyse the attention maps produced by ProbSA and compare them to those from other methods. We display the attention maps of a CT scan from RSNA in Fig. 5, a WSI from PANDA in Fig. 7, and a WSI from CAMELYON16 in Fig. 6. Additional attention maps, including two more CT scans, two WSIs from PANDA, and two WSIs from CAMELYON16, are provided in Appendix C. In total, we present attention maps for methods from each family across three representative positive bags from each dataset. Notably, the instance labels displayed are used strictly for evaluation purposes and were not available during training.

ProbSA is aligned with previous works. Previous works have shown that positive instances, i.e., those that are unhealthy, typically receive high attention values. This enables one to locate areas containing hemorrhage (in the case of RSNA) or cancerous regions (in the case of PANDA and CAMELYON16). Fig. 5, Fig. 6, and Fig. 7 show that the attention maps generated by ProbSA also exhibit this desired pattern, demonstrating performance comparable to other leading methods, including DTFD-MIL, PatchGCN, GTP, and CAMIL.

ProbSA reduces false positives. Other methods tends to produce undesirable medium-to-high intensity isolated red spots in healthy areas, falsely indicating unhealthy instances. A clear example of this phenomena is in Fig. 7, where PatchGCN and DTFD-MIL assign high attention to healthy patches. Other examples are Fig. 5, Fig. C.10, Fig. C.14, and Fig. C.15. Note that these spots do

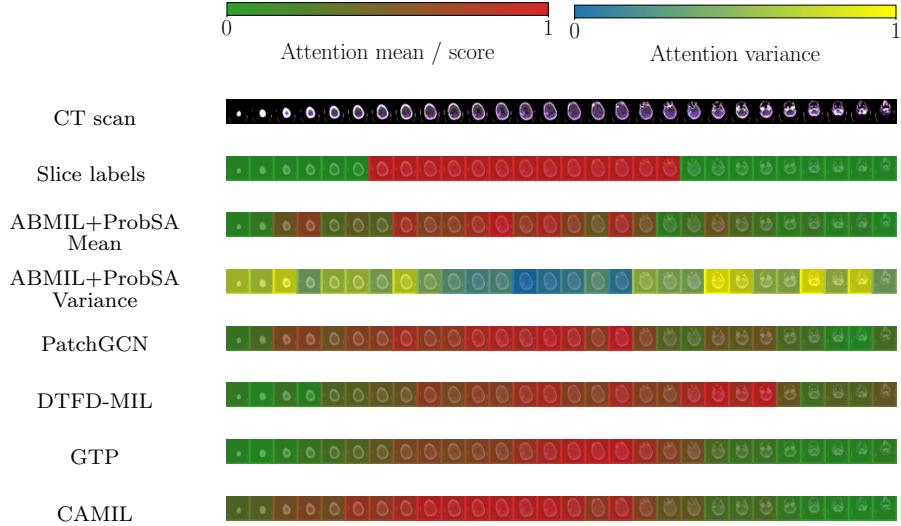


Figure 5: Attention maps in a CT scan from RSNA. The attention values have been normalized to ease visualization. The proposed ProbSA produces a very accurate attention map and flags false positives with high variance.

not appear on the maps generated by ProbSA. We attribute this behaviour to the training objective, which encourages the model to produce *smooth* attention maps, minimizing isolated high-attention regions in healthy areas.

Attention variance flags wrong predictions. In practice, there are situations where the attention maps are not accurate. For example, in Fig. 6 none of the considered methods completely detects the cancerous area. Unlike the rest of the methods, the novel probabilistic ProbSA includes a mechanism to measure uncertainty, and one would expect that these wrong predictions are flagged with high variance. Indeed, in Fig. 6 we observe that ProbSA assigns high variance to these incorrectly predicted instances. Fig. C.9 and Fig. C.12 show a similar behaviour: as the other competing methods, ProbSA wrongly assigns high attention values to healthy instances but, unlike those baselines, it flags out these instances through a high variance value.

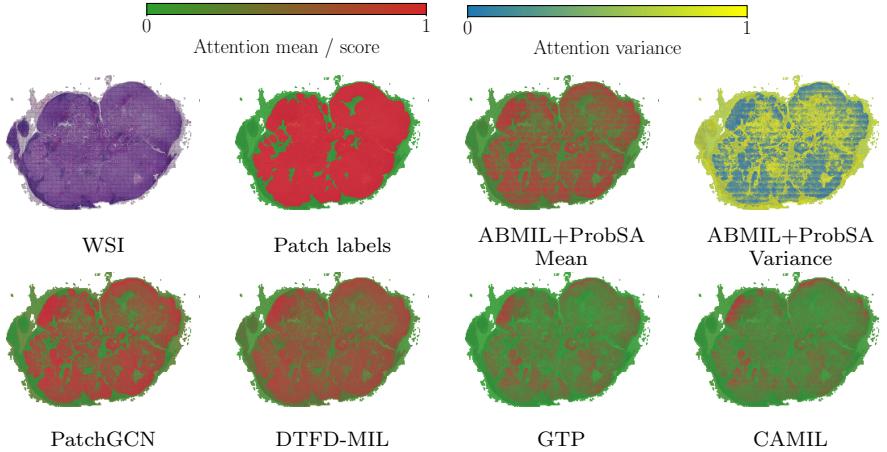


Figure 6: Attention maps in a WSI from CAMELYON16. The attention values have been normalized to ease visualization. While none of the methods is able to fully localize the cancerous area, the proposed ProbSA is the only capable of flagging its wrong predictions.

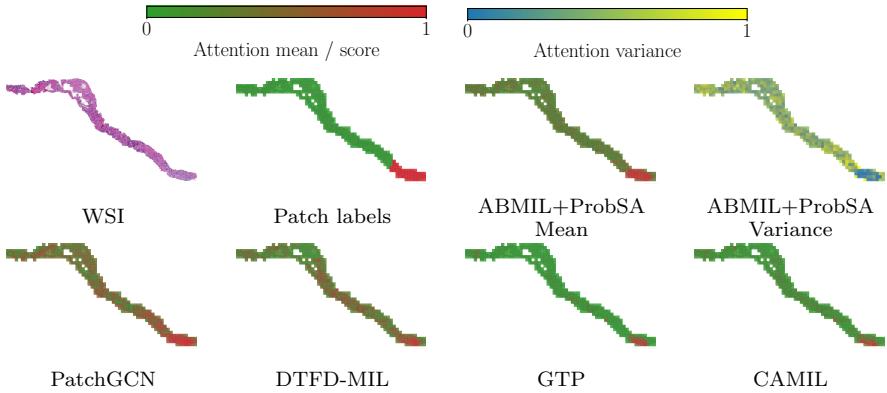


Figure 7: Attention maps in a WSI from PANDA. The attention values have been normalized to ease visualization. The proposed ProbSA produces fewer false positives than ABMIL, PatchGCN, and DTFD-MIL. Also, positive patches with low attention are flagged with high variance (see bottom of the WSI).

5. Conclusion and limitations

In this work, we have introduced ProbSA, a new attention-based deep MIL approach for medical imaging that accounts for both local and global interactions among instances in the bag. Local interactions, which refer to dependencies among neighbouring instances, are introduced by favouring smooth atten-

tion values within the bag. Global interactions, which are longer-range dependencies, are modelled using a Transformer encoder. Moreover, we propose a probabilistic formulation that estimates attention values through a probability distribution, instead of deterministically through a single numeric value. To the best of our knowledge, this is the first method that tackles both types of interactions in a probabilistic manner. We show enhanced predictive performance against current SOTA deep MIL approaches, as well as interpretability of the attention uncertainty maps provided by the probabilistic treatment.

There exist two main limitations to the proposed approach. First, we have only analyzed two possibilities for the variational attention distribution: a Dirac delta, which leads to a deterministic treatment, and a Gaussian distribution with a diagonal covariance matrix, which can be treated through the re-parametrization trick and is computationally tractable. However, more expressive variational distributions, possibly accounting for different modes, could be leveraged at a higher computational cost. Second, the proposed method does not solve the localization issues of current attention-based deep MIL approaches. Whereas the uncertainty maps are useful to highlight low-confidence regions, the mean of the attention does not accurately estimate the instance labels in every case, similar to existing methods.

In future work, alongside addressing the aforementioned limitations, we will explore alternative priors to enforce smoothness in the attention maps. However, other approaches different than the one used in this study may render an intractable KL divergence term, and thus they would require careful handling.

Acknowledgments

This work was supported by project PID2022-140189OB-C22 funded by MCIN / AEI / 10.13039 / 501100011033. Francisco M. Castro-Macías acknowledges FPU contract FPU21/01874 funded by Ministerio de Universidades. Pablo Morales-Álvarez acknowledges grant C-EXP-153-UGR23 funded by Consejería de Universidad, Investigación e Innovación and by the European Union

(EU) ERDF Andalusia Program 2021-2027. Funding for open access charge:
Universidad de Granada / CBUA.

References

- [1] T. G. Dietterich, R. H. Lathrop, T. Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, *Artificial intelligence* (1997).
- [2] A. H. Song, G. Jaume, D. F. Williamson, M. Y. Lu, A. Vaidya, T. R. Miller, F. Mahmood, Artificial intelligence for digital and computational pathology, *Nature Reviews Bioengineering* 1 (2023) 930–949.
- [3] M. Ilse, J. Tomczak, M. Welling, Attention-based deep multiple instance learning, in: *International Conference on Machine Learning*, 2018.
- [4] M. Y. Lu, D. F. Williamson, T. Y. Chen, R. J. Chen, M. Barbieri, F. Mahmood, Data-efficient and weakly supervised computational pathology on whole-slide images, *Nature biomedical engineering* (2021).
- [5] H. Zhang, Y. Meng, Y. Zhao, Y. Qiao, X. Yang, S. E. Coupland, Y. Zheng, Dtfd-mil: Double-tier feature distillation multiple instance learning for histopathology whole slide image classification, in: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18802–18812.
- [6] B. Li, Y. Li, K. W. Eliceiri, Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning, in: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14318–14328.
- [7] Z. Shao, H. Bian, Y. Chen, Y. Wang, J. Zhang, et al., Transmil: Transformer based correlated multiple instance learning for whole slide image classification, *Advances in neural information processing systems* (2021).
- [8] H. Li, F. Yang, Y. Zhao, X. Xing, J. Zhang, M. Gao, J. Huang, L. Wang, J. Yao, Dt-mil: deformable transformer for multi-instance learning on

- histopathological image, in: International Conference on Medical Image Computing and Computer Assisted Intervention, Springer, 2021.
- [9] O. Fourkioti, M. D. Vries, C. Bakal, CAMIL: Context-aware multiple instance learning for cancer detection and subtyping in whole slide images, in: International Conference on Learning Representations, 2024.
 - [10] B. Yang, C. Jiao, J. Wu, L. Li, Variational multiple-instance learning with embedding correlation modeling for hyperspectral target detection, IEEE Transactions on Neural Networks and Learning Systems (2024).
 - [11] Y. Zheng, R. H. Gindra, E. J. Green, E. J. Burks, M. Betke, J. E. Beane, V. B. Kolachalama, A graph-transformer for whole slide image classification, IEEE transactions on medical imaging 41 (2022).
 - [12] Y. Zhao, Z. Lin, K. Sun, Y. Zhang, J. Huang, L. Wang, J. Yao, Setmil: spatial encoding transformer-based multiple instance learning for pathological image analysis, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2022.
 - [13] F. M. Castro-Macías, P. Morales-Alvarez, Y. Wu, R. Molina, A. Katsaggelos, Sm: enhanced localization in multiple instance learning for medical imaging classification, in: The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024.
 - [14] R. Li, J. Yao, X. Zhu, Y. Li, J. Huang, Graph cnn for survival analysis on whole slide pathological images, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2018.
 - [15] R. J. Chen, M. Y. Lu, M. Shaban, C. Chen, et al., Whole slide images are 2d point clouds: Context-aware survival prediction using patch-based graph convolutional networks, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2021.
 - [16] Y. Wu, F. M. Castro-Macías, P. Morales-Álvarez, R. Molina, A. K. Katsaggelos, Smooth attention for deep multiple instance learning: Application

- to ct intracranial hemorrhage detection, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2023.
- [17] D. Zhou, O. Bousquet, T. Lal, J. Weston, B. Schölkopf, Learning with local and global consistency, Advances in neural information processing systems (2003).
 - [18] B. D. Ripley, Spatial statistics (1981).
 - [19] C. M. Bishop, N. M. Nasrabadi, Pattern recognition and machine learning, volume 4, Springer, 2006.
 - [20] D. P. Kingma, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).
 - [21] G. B. Arfken, H. J. Weber, F. E. Harris, Mathematical methods for physicists: a comprehensive guide, Academic press, 2011.
 - [22] G. B. Folland, Fourier analysis and its applications, volume 4, American Mathematical Soc., 2009.
 - [23] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, A. Lerchner, beta-vae: Learning basic visual concepts with a constrained variational framework., ICLR 3 (2017).
 - [24] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, L. Carin, Cyclical annealing schedule: A simple approach to mitigating kl vanishing, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019.
 - [25] C. M. Bishop, H. Bishop, Deep Learning: Foundations and Concepts, 2023.
 - [26] A. E. Flanders, L. M. Prevedello, G. Shih, S. S. Halabi, J. Kalpathy-Cramer, R. Ball, et al., Construction of a machine learning dataset through collaboration: the rsna 2019 brain ct hemorrhage challenge, Radiology: Artificial Intelligence (2020).

- [27] W. Bulten, K. Kartasalo, P.-H. C. Chen, P. Ström, H. Pinckaers, K. Nagpal, Y. Cai, D. F. Steiner, et al., Artificial intelligence for diagnosis and gleason grading of prostate cancer: the panda challenge, *Nature medicine* (2022).
- [28] B. E. Bejnordi, M. Veta, P. J. Van Diest, B. Van Ginneken, N. Karssemeijer, G. Litjens, J. A. Van Der Laak, et al., Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer, *Jama* (2017).
- [29] J. Silva-Rodriguez, A. Colomer, J. Dolz, V. Naranjo, Self-learning for weakly supervised gleason grading of local patterns, *IEEE journal of biomedical and health informatics* 25 (2021) 3094–3104.
- [30] M. Kang, H. Song, S. Park, D. Yoo, S. Pereira, Benchmarking self-supervised learning on diverse pathology datasets, in: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [31] X. Wang, J. Zhao, E. Marostica, W. Yuan, J. Jin, J. Zhang, R. Li, H. Tang, K. Wang, Y. Li, et al., A pathology foundation model for cancer diagnosis and prognosis prediction, *Nature* (2024) 1–9.
- [32] Q. Ren, Y. Zhao, B. He, B. Wu, S. Mai, et al., Iibmil: Integrated instance-level and bag-level multiple instances learning with label disambiguation for pathological image analysis, in: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2023.

Appendix A. Variational inference with a Dirac delta

We provide a derivation of the training objective for the deterministic variant of the proposed method, see Eq. 16. The fundamental idea here is to approximate the Dirac delta using a sequence of continuous densities of compact support. For simplicity, we assume bags of constant size $N \in \mathbb{N}$. Our derivation is based on the following result.

Preliminary result. Let $u: \mathbb{R}^N \rightarrow \mathbb{R}$ be a continuous function of compact support such that $\int_{\mathbb{R}^N} u(\mathbf{x}) d\mathbf{x} = 1$. For $\varepsilon > 0$, we define

$$u_\varepsilon(\mathbf{x}) = \varepsilon^{-N} u(\varepsilon^{-1}\mathbf{x}), \quad \forall \mathbf{x} \in \mathbb{R}^N. \quad (\text{A.1})$$

Let $\mathbf{a} \in \mathbb{R}^N$. For any continuous function $g: \mathbb{R}^N \rightarrow \mathbb{R}$, it holds that

$$\lim_{\varepsilon \rightarrow 0^+} \int_{\mathbb{R}^N} u_\varepsilon(\mathbf{x} - \mathbf{a}) g(\mathbf{x}) d\mathbf{x} = g(\mathbf{a}). \quad (\text{A.2})$$

See [22, Theorem 7.3] for a proof. This expression corresponds to the definition of the Dirac delta using a (continuous and compactly supported) approximation to the identity [22]. It is usually written informally as

$$\int_{\mathbb{R}^N} \delta(\mathbf{x} - \mathbf{a}) g(\mathbf{x}) d\mathbf{x} = g(\mathbf{a}). \quad (\text{A.3})$$

Derivation of Eq. 16. We are now ready to derive Eq. 16. Let $\mu: \mathbb{R}^{N \times D} \rightarrow \mathbb{R}^N$ a neural network parameterizing the attention values. Let u_ε defined as in Eq. A.1. We consider a family $\{q_\varepsilon\}_{\varepsilon > 0}$ of variational posteriors defined as

$$q_\varepsilon(\mathbf{f} | \mathbf{X}) = u_\varepsilon(\mathbf{f} - \mu(\mathbf{X})). \quad (\text{A.4})$$

Intuitively, $\{q_\varepsilon\}_{\varepsilon > 0}$ converges to the variational posterior defined using the Dirac delta in Eq. 15. Let us write the ELBO for this family,

$$\text{ELBO}_\varepsilon = \sum_{b=1}^B \{\mathbb{E}_{q_\varepsilon(\mathbf{f}_b | \mathbf{X}_b)} [\log p(Y_b | \mathbf{X}_b, \mathbf{f}_b)] + \mathbb{E}_{q_\varepsilon(\mathbf{f}_b | \mathbf{X}_b)} [\log p(\mathbf{f}_b)] + \quad (\text{A.5})$$

$$- \mathbb{E}_{q_\varepsilon(\mathbf{f}_b | \mathbf{X}_b)} [\log q_\varepsilon(\mathbf{f}_b | \mathbf{X}_b)]\} \quad (\text{A.6})$$

We are interested in optimizing this ELBO with respect to the parameters of our neural network in the limit $\varepsilon \rightarrow 0^+$. For the last term, after a change of

variable, we have

$$\mathbb{E}_{q_\varepsilon(\mathbf{f}_b|\mathbf{X}_b)} [\log q_\varepsilon(\mathbf{f}_b | \mathbf{X}_b)] = \int_{\mathbb{R}^N} u_\varepsilon(\mathbf{x}) \log u_\varepsilon(\mathbf{x}) d\mathbf{x}. \quad (\text{A.7})$$

Note that this term does not depend on the parameters of the neural network, and is well defined for $\varepsilon > 0$. However, its limit when $\varepsilon \rightarrow 0^+$ may not be well defined. This is not a problem for us, since we are only interested in those terms that depend on the parameters of the neural network, i.e.,

$$\widetilde{\text{ELBO}}_\varepsilon = \sum_{b=1}^B \left\{ \mathbb{E}_{q_\varepsilon(\mathbf{f}_b|\mathbf{X}_b)} [\log p(Y_b | \mathbf{X}_b, \mathbf{f}_b)] + \mathbb{E}_{q_\varepsilon(\mathbf{f}_b|\mathbf{X}_b)} [\log p(\mathbf{f}_b)] \right\}. \quad (\text{A.8})$$

Finally, we take the limit when $\varepsilon \rightarrow 0^+$ and apply the preliminary result in Eq. A.2, obtaining the desired optimization objective, see Eq. 16.

Appendix B. Transformer encoder

We provide more details about the Transformer encoder introduced in Sec. 3.2. At the core of this module is the self-attention mechanism, which is given an input bag $\mathbf{X} \in \mathbb{R}^{N \times D}$ and outputs a new bag $\mathbf{Y} = \text{SelfAttention}(\mathbf{X}) \in \mathbb{R}^{N \times D_v}$ transformed according to

$$\text{SelfAttention}(\mathbf{X}) = \text{Softmax} \left(q(\mathbf{X}) k(\mathbf{X})^\top \right) v(\mathbf{X}), \quad (\text{B.1})$$

where $q, k: \mathbb{R}^{N \times D} \rightarrow \mathbb{R}^{N \times D_{qk}}$, and $v: \mathbb{R}^{N \times D} \rightarrow \mathbb{R}^{N \times D_v}$ are the standard queries, keys, and values in the dot product self-attention [25]. Each layer of the Transformer encoder leverages multiple heads in the self-attention mechanism, along with skip-connections and pre-layer normalization [25],

$$\text{TransformerEnc}(\mathbf{X}) = \mathbf{Y}^L, \quad (\text{B.2})$$

$$\mathbf{Y}^0 = \mathbf{X}, \quad (\text{B.3})$$

$$\mathbf{Z}^{l+1} = \mathbf{Y}^l + \text{SelfAttention}(\text{LayerNorm}(\mathbf{Y}^l)), \quad (\text{B.4})$$

$$\mathbf{Y}^{l+1} = \mathbf{Z}^{l+1} + \text{MLP}(\text{LayerNorm}(\mathbf{Z}^{l+1})), \quad (\text{B.5})$$

where $l \in \{0, \dots, L-1\}$, and MLP denotes a two-layer perceptron. As explained in Sec. 4, in this work we use $D_{qk} = D_v = 512$, $L = 2$ and 8 attention heads.

Appendix C. Additional tables and figures

Computational overhead. ProbSA incorporates local interactions into the loss function through the bag adjacency matrices. These are instantiated as sparse tensors, reducing memory usage and enabling efficient matrix-vector operations. As shown in Table C.4, the overhead is comparable to that of other widely used methods. Relative to the baseline on top of which they are implemented, training time increases slightly on RSNA and PANDA, and more significantly on CAMELYON16, due to the larger size of its WSIs. Nonetheless, the overhead remains manageable, and our method scales effectively to large datasets such as CAMELYON16.

Model	Time per step (ms)			Time per step (ms)		
	RSNA	PANDA	CAMELYON16	RSNA	PANDA	CAMELYON16
ABMIL+ProbSA($\Sigma = 0$)	2.769 \pm 0.280	2.966 \pm 0.992	15.334 \pm 13.077	3.374 \pm 0.315	3.562 \pm 0.853	47.609 \pm 52.472
ABMIL+ProbSA($\Sigma = \text{Diag}$)	3.776 \pm 0.377	3.821 \pm 1.315	16.008 \pm 12.469	4.49 \pm 0.452	4.091 \pm 1.118	49.172 \pm 53.664
ABMIL	0.997 \pm 0.132	1.035 \pm 0.129	2.094 \pm 0.653	2.182 \pm 0.199	2.169 \pm 0.201	3.65 \pm 0.805
CLAM	3.832 \pm 0.407	4.896 \pm 4.035	5.401 \pm 7.563	10.596 \pm 0.643	11.153 \pm 0.745	13.960 \pm 4.380
DSMIL	1.692 \pm 0.275	2.468 \pm 3.280	3.051 \pm 5.238	24.321 \pm 1.528	36.289 \pm 2.016	307.732 \pm 103.096
PatchGCN	2.356 \pm 0.215	3.132 \pm 3.636	5.790 \pm 12.516	6.501 \pm 0.311	8.059 \pm 0.384	11.794 \pm 8.766
DTFD-MIL	5.010 \pm 0.364	6.373 \pm 4.265	7.434 \pm 13.055	8.718 \pm 0.566	19.087 \pm 1.722	61.792 \pm 51.316
VAMIL				6.172 \pm 0.374	6.766 \pm 0.883	8.685 \pm 9.011
CAMIL				12.231 \pm 0.728	16.880 \pm 0.419	58.387 \pm 2.148

Table C.4: Time per training step (in milliseconds). A training step is defined as a single forward-backward pass.

		RSNA		PANDA		CAMELYON16			
Model	λ	AUROC (\uparrow)	F1 (\uparrow)	AUROC (\uparrow)	F1 (\uparrow)	AUROC (\uparrow)	F1 (\uparrow)	Rank (\downarrow)	
ABMIL+ProbSA $\Sigma = 0$	0.0	87.924 _{1.179}	78.317 _{2.531}	97.843 _{0.201}	95.060 _{0.435}	97.454 _{1.074}	90.756 _{2.483}	4.000 _{1.265}	
	0.1	88.631 _{0.464}	79.479 _{2.294}	98.190 _{0.045}	95.398 _{0.252}	98.628 _{0.468}	91.122 _{1.762}	2.500 _{1.643}	
	0.5	89.384 _{0.612}	81.167 _{1.951}	97.557 _{0.114}	94.451 _{0.299}	97.459 _{1.135}	92.113 _{1.941}	3.167 _{0.753}	
	1.0	89.950 _{0.990}	82.509 _{2.572}	97.146 _{0.178}	93.865 _{0.254}	96.888 _{1.555}	92.360 _{1.648}	3.333 _{1.862}	
	cyclical	90.189 _{0.482}	83.168 _{1.259}	97.896 _{0.090}	94.973 _{0.140}	97.633 _{0.618}	91.307 _{1.053}	2.000 _{0.894}	
ABMIL+ProbSA $\Sigma = \text{Diag}$	0.0	87.924 _{1.179}	78.317 _{2.531}	97.843 _{0.201}	95.060 _{0.435}	97.454 _{1.074}	90.756 _{2.483}	4.000 _{1.265}	
	0.1	88.957 _{0.575}	80.409 _{1.211}	98.178 _{0.048}	95.559 _{0.184}	98.571 _{0.389}	92.176 _{1.347}	2.167 _{1.472}	
	0.5	90.089 _{1.029}	82.297 _{1.947}	97.664 _{0.067}	94.643 _{0.184}	97.173 _{1.884}	91.452 _{2.825}	3.667 _{0.516}	
	1.0	90.666 _{0.877}	82.820 _{1.461}	97.121 _{0.138}	93.836 _{0.242}	96.949 _{1.111}	91.659 _{1.263}	3.500 _{1.761}	
	cyclical	90.231 _{0.401}	83.230 _{1.719}	98.094 _{0.069}	95.274 _{0.236}	97.885 _{1.259}	92.351 _{2.385}	1.667 _{0.516}	
T-ABMIL+ProbSA $\Sigma = 0$	0.0	91.083 _{0.978}	84.083 _{1.962}	98.014 _{0.226}	95.289 _{0.322}	97.673 _{0.695}	91.826 _{3.638}	3.500 _{1.225}	
	0.1	91.141 _{1.023}	83.023 _{2.429}	98.046 _{0.176}	95.378 _{0.190}	98.017 _{0.416}	93.193 _{3.426}	2.833 _{1.329}	
	0.5	91.079 _{1.171}	83.469 _{2.720}	98.064 _{0.135}	95.457 _{0.115}	98.304 _{0.412}	92.676 _{0.632}	2.667 _{1.506}	
	1.0	91.072 _{1.113}	83.580 _{2.784}	98.007 _{0.146}	95.254 _{0.305}	98.068 _{1.220}	93.131 _{1.753}	3.667 _{0.816}	
	cyclical	91.781 _{1.200}	84.591 _{2.535}	97.974 _{0.156}	95.213 _{0.072}	98.418 _{1.104}	94.220 _{2.168}	2.333 _{2.066}	
T-ABMIL+ProbSA $\Sigma = \text{Diag}$	0.0	91.083 _{0.978}	84.083 _{1.962}	98.014 _{0.226}	95.289 _{0.322}	97.673 _{0.695}	91.826 _{3.638}	2.500 _{1.975}	
	0.1	90.812 _{1.065}	83.260 _{2.006}	97.928 _{0.109}	95.211 _{0.216}	97.934 _{0.572}	93.834 _{1.816}	3.833 _{0.753}	
	0.5	90.808 _{1.015}	83.226 _{2.144}	97.974 _{0.187}	95.266 _{0.194}	98.486 _{1.147}	94.657 _{2.036}	3.000 _{1.414}	
	1.0	90.883 _{1.043}	82.964 _{2.064}	97.929 _{0.199}	95.198 _{0.325}	98.061 _{1.352}	92.653 _{2.638}	3.833 _{1.169}	
	cyclical	90.814 _{1.683}	83.683 _{3.669}	97.988 _{0.117}	95.339 _{0.249}	98.133 _{0.828}	94.732 _{3.071}	1.833 _{0.753}	

Table C.5: Classification results (mean and standard deviation from five independent runs) using different values for the balancing hyperparameter λ . The best in each group is in bold. (\downarrow)/(\uparrow) means lower/higher is better. The optimal choice of λ depends on the variant and the dataset. Setting $\lambda = \text{cyclical}$ provides the best rank for every variant.

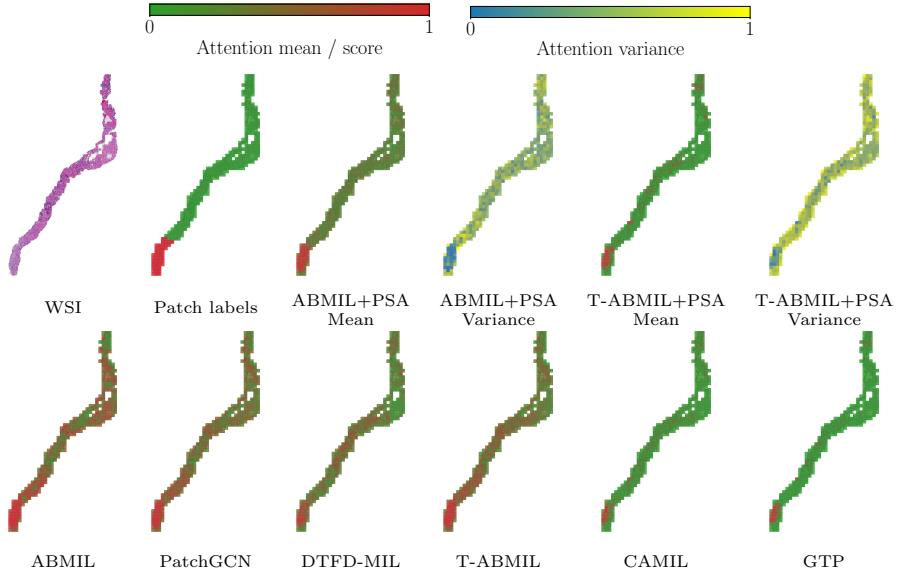


Figure C.8: Attention maps in a WSI from PANDA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

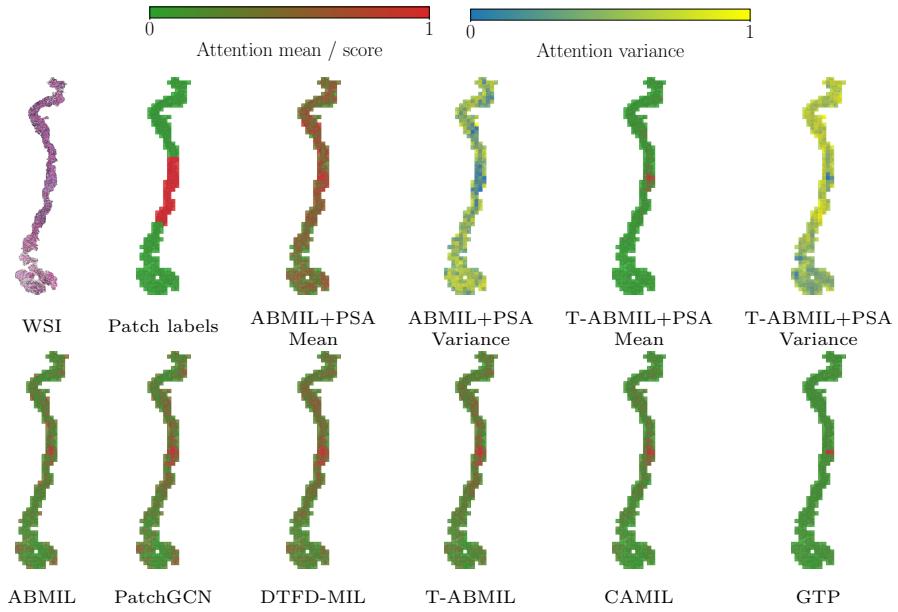


Figure C.9: Attention maps in a WSI from PANDA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

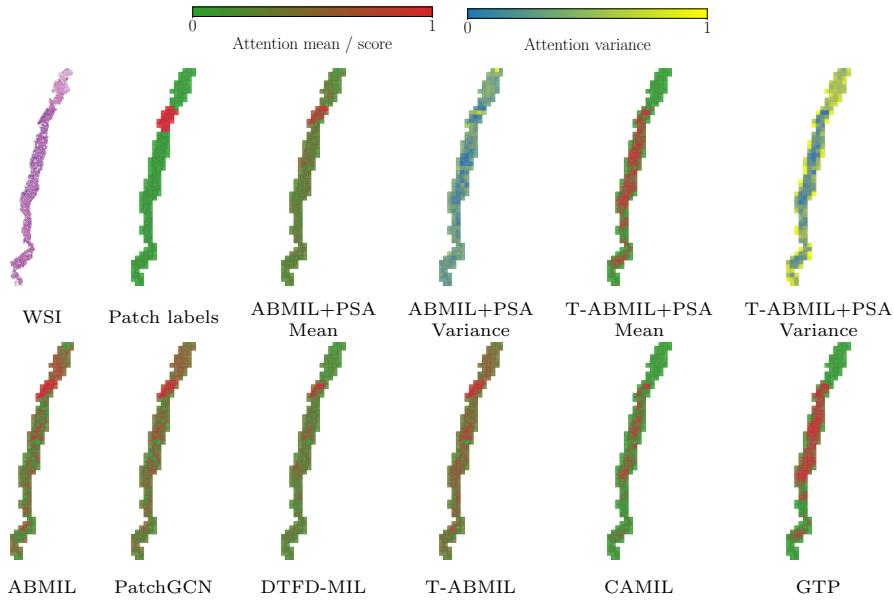


Figure C.10: Attention maps in a WSI from PANDA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

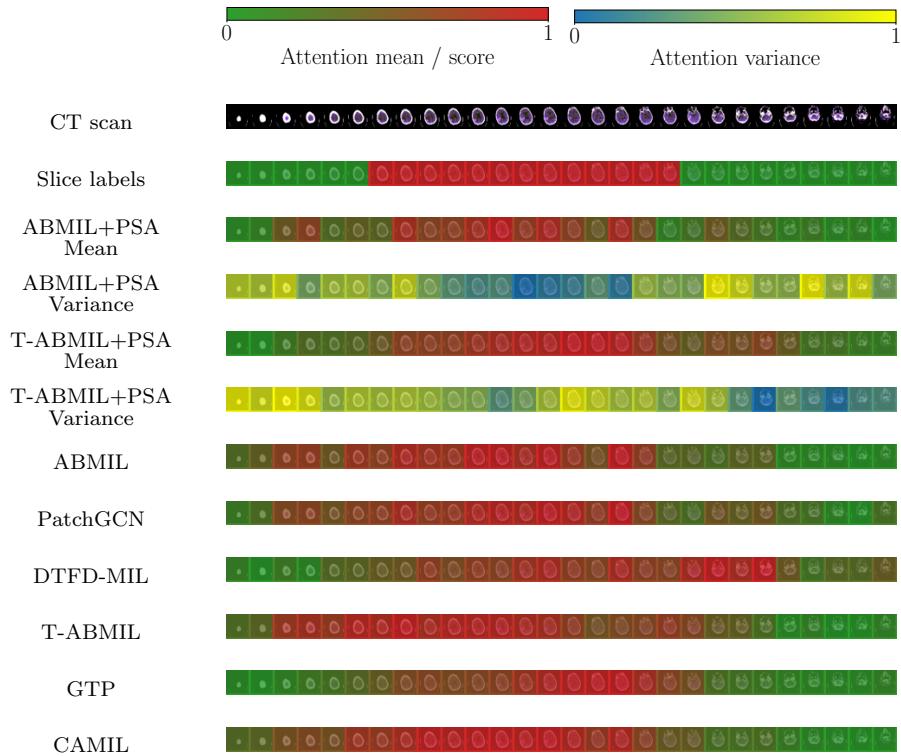


Figure C.11: Attention maps in a CT scan from RSNA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

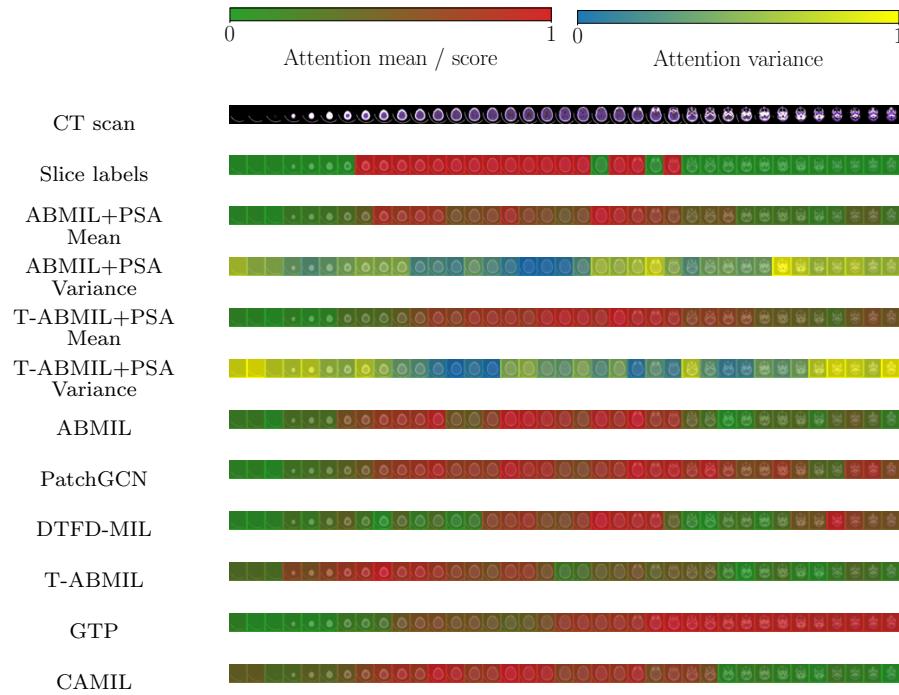


Figure C.12: Attention maps in a CT scan from RSNA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

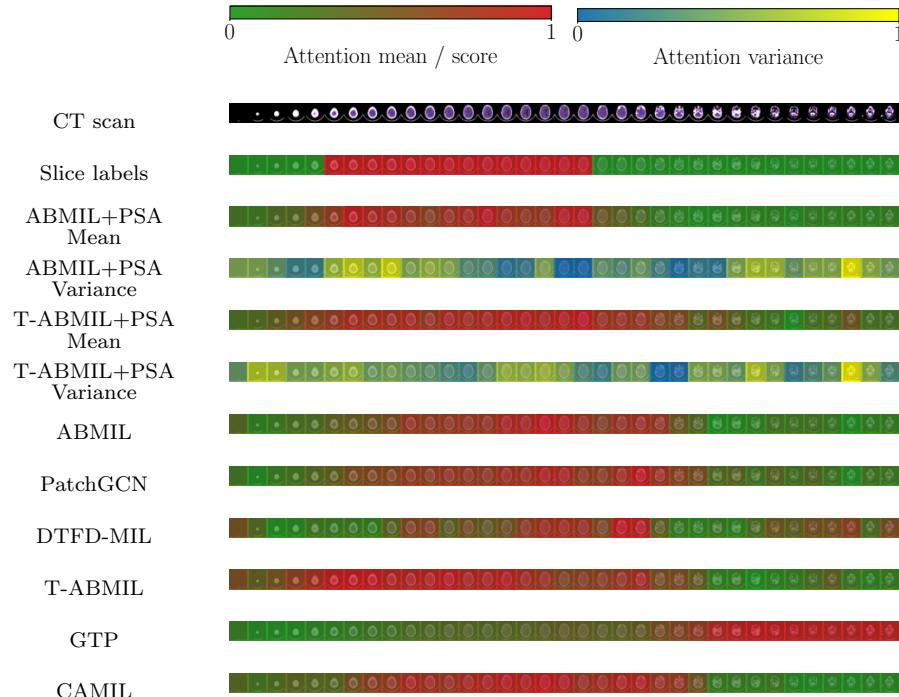


Figure C.13: Attention maps in a CT scan from RSNA. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

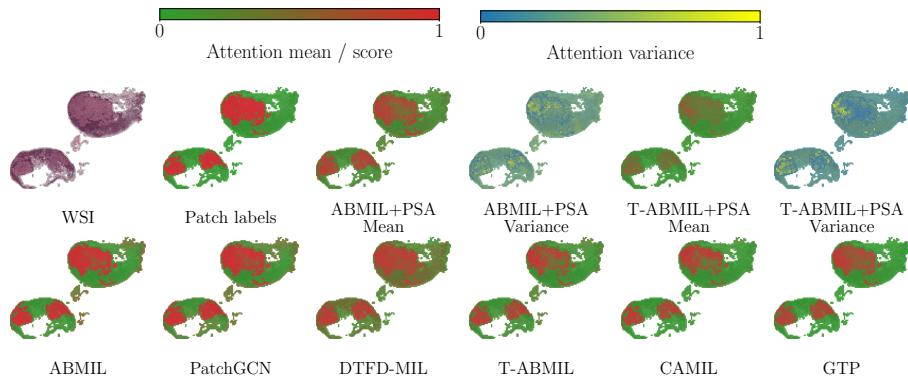


Figure C.14: Attention maps in a WSI from CAMELYON16. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

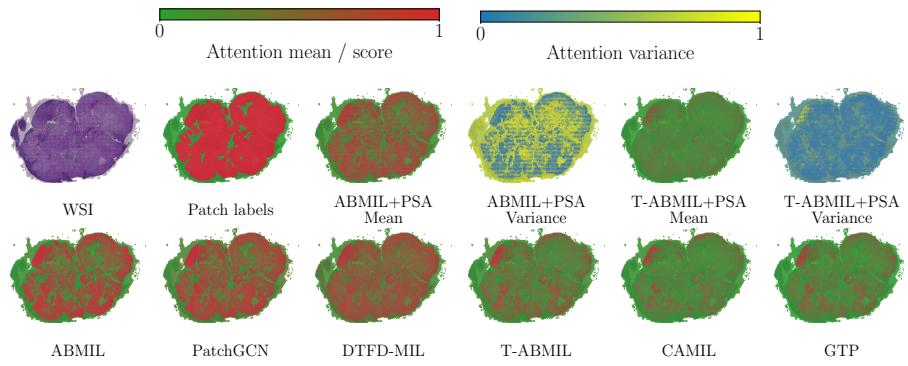


Figure C.15: Attention maps in a WSI from CAMELYON16. The attention values have been normalized to ease visualization. PSA stands for ProbSA.

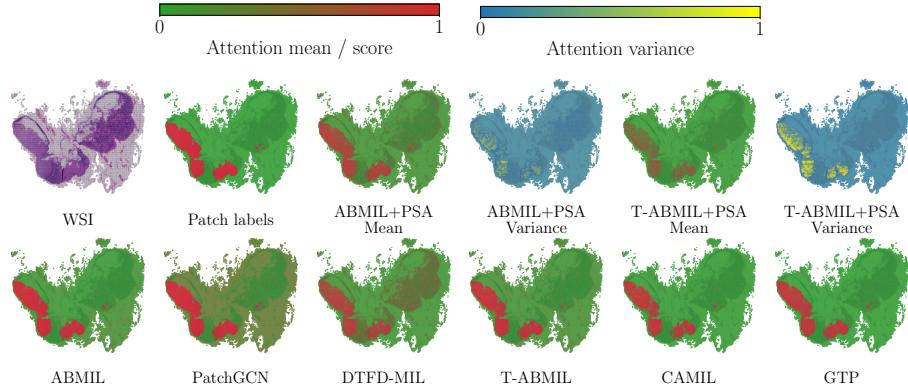


Figure C.16: Attention maps in a WSI from CAMELYON16. The attention values have been normalized to ease visualization. PSA stands for ProbSA.