6. Appendix

6.1. Deep MIL approaches

In Figure 6 we present three deep MIL approaches discussed in the paper.

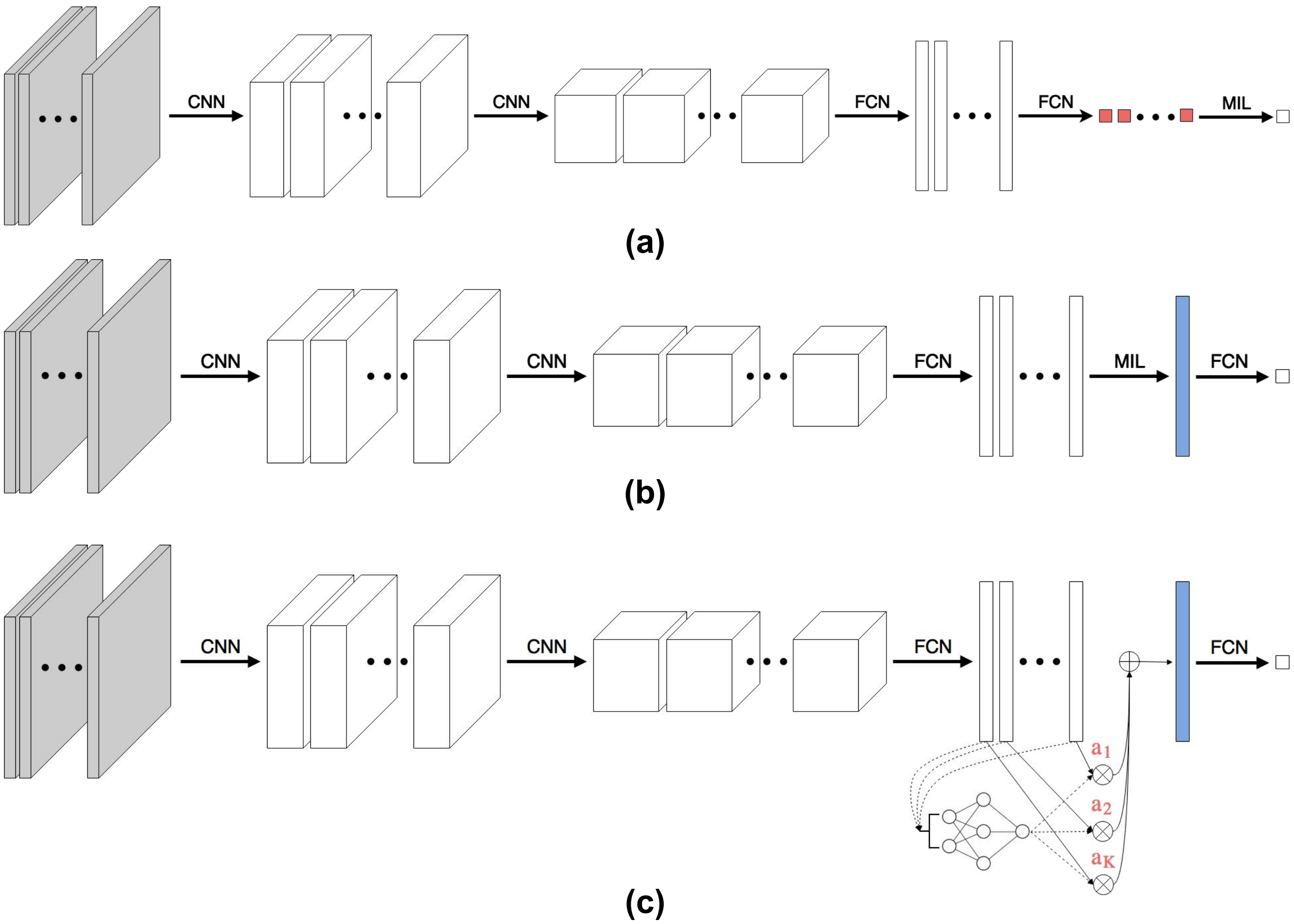

Figure 6. Deep MIL approaches: (a) the instance-based approach, (b) the embedding-based approach, (c) the proposed approach with the attention mechanism as the MIL pooling. Red color corresponds to instance scores, blue color depicts a bag vector representation. *Best viewed in color.*

6.2. Code

The implementation of our methods is available online at `https://github.com/AMLab-Amsterdam/AttentionDeepMIL`. All experiments were run on NVIDIA TITAN X Pascal with a batch size of 1 (= 1 bag) for all datasets.

6.3. Classical MIL datasets

Additional details In Table 1 a general description of the five benchmark MIL datasets used in the experiments is given. In Tables 5 and 6 we present architectures of the embedding-based and the instance-based models, respectively. We denote a fully-connected layer by ’fc’ and the number of output hidden units is provided after a dash. The ReLU non-linearity was used. In Table 7 the details of the optimization (learning) procedure are given. We provide values of hyperparameters determined by the model selection procedure for which the highest validation performance was achieved.

Table 4. Overview of classical MIL datasets.

Dataset	# of bags	# of instances	# of features
Musk1	92	476	166
Musk2	102	6598	166
Tiger	200	1220	230
Fox	200	1302	230
Elephant	200	1391	230

Table 5. Classical MIL datasets: The embedding-based model architecture (Wang et al., 2016).

Layer	Type
1	fc-256 + ReLU
2	dropout
3	fc-128 + ReLU
4	dropout
5	fc-64 + ReLU
6	dropout
7	mil-max/mil-mean/mil-attention-64
8	fc-1 + sigm

Table 6. Classical MIL datasets: The instance-based model architecture (Wang et al., 2016).

Layer	Type
1	fc-256 + ReLU
2	dropout
3	fc-128 + ReLU
4	dropout
5	fc-64 + ReLU
6	dropout
7	fc-1 + sigm
8	mil-max/mil-mean

Table 7. Classical MIL datasets: The optimization procedure details (Wang et al., 2016).

Experiment	Optimizer	Momentum	Learning rate	Weight decay	Epochs	Stopping criteria
Musk1	SGD	0.9	0.0005	0.005	100	lowest validation error and loss
Musk2	SGD	0.9	0.0005	0.03	100	lowest validation error and loss
Tiger	SGD	0.9	0.0001	0.01	100	lowest validation error and loss
Fox	SGD	0.9	0.0005	0.005	100	lowest validation error and loss
Elephant	SGD	0.9	0.0001	0.005	100	lowest validation error and loss

6.4. MNIST-bags

Additional details In Tables 8 and 9 we present architectures of the embedding-based and the instance-based models for MNIST-BAGS, respectively. We denote a convolutional layer by 'conv', in brackets we provide kernel size, stride and padding, and the number of kernels is provided after a dash. The convolutional max-pooling layer is denoted by 'maxpool' and the pooling size is given in brackets. The ReLU non-linearity was used. In Table 10 the details of the optimization (learning) procedure for deep MIL approach are given. The details of the SVM are given in Table 11. We provide values of hyperparameters determined by the model selection procedure for which the highest validation performance was achieved.

Table 8. MNIST-bags: The embedding-based model architecture (LeCun et al., 1998).

Layer	Type
1	conv(5,1,0)-20 + ReLU
2	maxpool(2,2)
3	conv(5,1,0)-50 + ReLU
4	maxpool(2,2)
5	fc-500 + ReLU
6	mil-max/mil-mean/mil-attention-128
7	fc-1 + sigm

Table 9. MNIST-bags: The instance-based model architecture (LeCun et al., 1998).

Layer	Type
1	conv(5,1,0)-20 + ReLU
2	maxpool(2,2)
3	conv(5,1,0)-50 + ReLU
4	maxpool(2,2)
5	fc-500 + ReLU
6	fc-1 + sigm
7	mil-max/mil-mean

Table 10. MNIST-bags: The optimization procedure details.

Experiment	Optimizer	β_1, β_2	Learning rate	Weight decay	Epochs	Stopping criteria
All	Adam	0.9, 0.999	0.0005	0.0001	200	lowest validation error+loss

Table 11. MNIST-bags: SVM configuration.

Model	Features	Kernel	C	γ	Max iterations
MI-SVM	Raw pixel values	RBF	5	0.0005	200

Additional results In Tables 12, 13 and 14 we present the test AUC value for 10, 50 and 100 instances on average per a bag, respectively.

In Figure 7 a negative bag is presented. In Figure 8 a positive bag with a single ’9’ is given. In Figure 9 a positive bag with multiple ’9’s is presented. In all figures attention weights are provided and in the case of positive bags a red rectangle highlights positive instances.

Table 12. The test AUC for MNIST-bags with on average 10 instances per bag for different numbers of training bags.

# of training bags	50	100	150	200	300	400	500
Instance+max	0.553 ± 0.053	0.745 ± 0.100	0.960 ± 0.004	0.979 ± 0.001	0.984 ± 0.001	0.986 ± 0.001	0.986 ± 0.001
Instance+mean	0.663 ± 0.014	0.676 ± 0.012	0.694 ± 0.010	0.694 ± 0.017	0.709 ± 0.020	0.693 ± 0.023	0.712 ± 0.018
MI-SVM	0.697 ± 0.054	0.851 ± 0.009	0.862 ± 0.008	0.898 ± 0.014	0.926 ± 0.004	0.942 ± 0.002	0.948 ± 0.002
Embedded+max	0.713 ± 0.016	0.914 ± 0.011	0.954 ± 0.005	0.968 ± 0.001	0.980 ± 0.001	0.981 ± 0.003	0.986 ± 0.002
Embedded+mean	0.695 ± 0.026	0.841 ± 0.027	0.926 ± 0.004	0.953 ± 0.004	0.974 ± 0.002	0.980 ± 0.001	0.984 ± 0.002
Attention	0.768 ± 0.054	0.948 ± 0.007	0.949 ± 0.006	0.970 ± 0.003	0.980 ± 0.000	0.982 ± 0.001	0.986 ± 0.001
Gated Attention	0.753 ± 0.054	0.916 ± 0.013	0.955 ± 0.003	0.974 ± 0.002	0.980 ± 0.004	0.983 ± 0.002	0.987 ± 0.001

Table 13. The test AUC for MNIST-bags with on average 50 instances per bag for different numbers of training bags.

# of training bags	50	100	150	200	300	400	500
Instance+max	0.576 ± 0.059	0.715 ± 0.096	0.937 ± 0.045	0.992 ± 0.002	0.994 ± 0.001	0.997 ± 0.001	0.997 ± 0.001
Instance+mean	0.737 ± 0.014	0.744 ± 0.029	0.824 ± 0.012	0.813 ± 0.030	0.722 ± 0.021	0.728 ± 0.017	0.798 ± 0.011
MI-SVM	0.824 ± 0.067	0.946 ± 0.004	0.959 ± 0.002	0.967 ± 0.002	0.975 ± 0.001	0.976 ± 0.001	0.979 ± 0.001
Embedded+max	0.872 ± 0.039	0.984 ± 0.005	0.992 ± 0.001	0.996 ± 0.001	0.996 ± 0.001	0.997 ± 0.001	0.997 ± 0.001
Embedded+mean	0.841 ± 0.013	0.906 ± 0.046	0.983 ± 0.005	0.992 ± 0.001	0.996 ± 0.001	0.997 ± 0.001	0.997 ± 0.001
Attention	0.967 ± 0.010	0.982 ± 0.003	0.990 ± 0.002	0.993 ± 0.002	0.989 ± 0.003	0.994 ± 0.001	0.995 ± 0.001
Gated Attention	0.920 ± 0.042	0.977 ± 0.006	0.993 ± 0.003	0.991 ± 0.002	0.994 ± 0.002	0.995 ± 0.001	0.996 ± 0.001

Table 14. The test AUC for MNIST-bags with on average 100 instances per bag for different numbers of training bags.

# of training bags	50	100	150	200	300	400	500
Instance+max	0.543 ± 0.054	0.804 ± 0.107	0.899 ± 0.086	0.999 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Instance+mean	0.842 ± 0.023	0.855 ± 0.025	0.824 ± 0.014	0.896 ± 0.037	0.859 ± 0.029	0.899 ± 0.012	0.868 ± 0.016
MI-SVM	0.871 ± 0.060	0.991 ± 0.002	0.994 ± 0.002	0.996 ± 0.001	0.997 ± 0.001	0.998 ± 0.001	0.998 ± 0.001
Embedded+max	0.977 ± 0.009	0.999 ± 0.001	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Embedded+mean	0.959 ± 0.010	0.990 ± 0.003	0.998 ± 0.001	0.900 ± 0.089	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Attention	0.996 ± 0.001	0.998 ± 0.001	0.999 ± 0.000	0.998 ± 0.001	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Gated Attention	0.998 ± 0.001	0.999 ± 0.000	0.998 ± 0.001	0.998 ± 0.001	0.999 ± 0.000	1.000 ± 0.000	1.000 ± 0.000

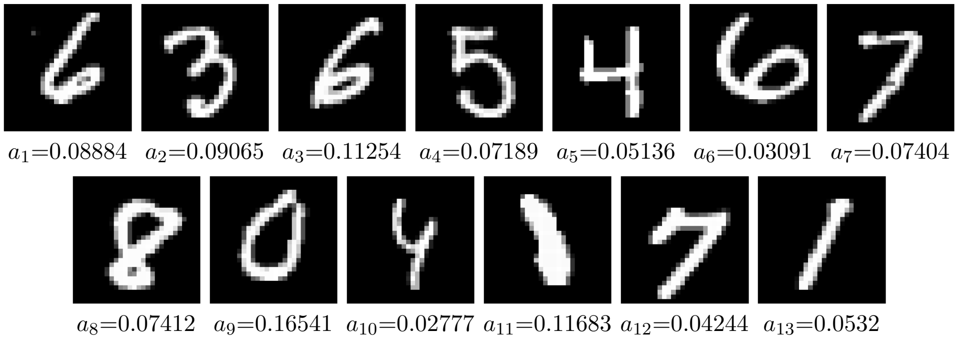

Figure 7. Example of attention weights for a negative bag.

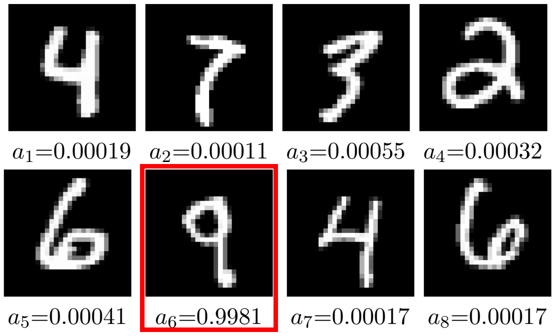

Figure 8. Example of attention weights for a positive bag containing a single ’9’.

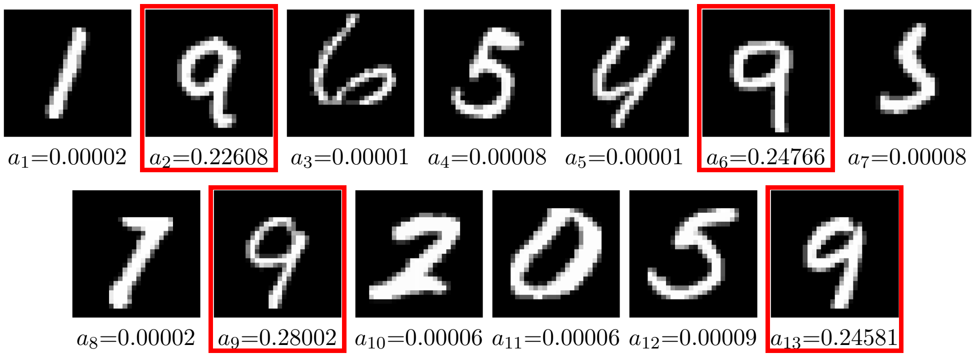

Figure 9. Example of attention weights for a positive bag containing multiple ’9’s.

6.5. Histopathology datasets

Data augmentation We randomly adjust the amount of H&E by decomposing the RGB color of the tissue into the H&E color space (Ruifrok & Johnston, 2001), followed by multiplying the magnitude of H&E for a pixel by two i.i.d. Gaussian random variables with expectation equal to one. We randomly rotate and mirror every patch. Lastly, we perform color normalization on every patch.

Additional details In Tables 15 and 16 we present architectures of the embedding-based and the instance-based models for histopathology datasets, respectively. In Table 17 the details of the optimization (learning) procedure for deep MIL approach are given. We provide values of hyperparameters determined by the model selection procedure for which the highest validation performance was achieved.

Table 15. Histopathology: The embedding-based model architecture (Sirinukunwattana et al., 2016).

Layer	Type
1	conv(4,1,0)-36 + ReLU
2	maxpool(2,2)
3	conv(3,1,0)-48 + ReLU
4	maxpool(2,2)
5	fc-512 + ReLU
6	dropout
7	fc-512 + ReLU
8	dropout
9	mil-max/mil-mean/mil-attention-128
10	fc-1 + sigm

Table 16. Histopathology: The instance-based model architecture (Sirinukunwattana et al., 2016).

Layer	Type
1	conv(4,1,0)-36 + ReLU
2	maxpool(2,2)
3	conv(3,1,0)-48 + ReLU
4	maxpool(2,2)
5	fc-512 + ReLU
6	dropout
7	fc-512 + ReLU
8	dropout
9	fc-1 + sigm
10	mil-max/mil-mean

Table 17. Histopathology: The optimization procedure details.

Experiment	Optimizer	β_1, β_2	Learning rate	Weight decay	Epochs	Stopping criteria
All	Adam	0.9, 0.999	0.0001	0.0005	100	lowest validation error+loss

Additional results In Figures 10, 11 and 12 five images are presented: (a) a full H&E image, (b) all patches containing cells, (c) positive patches, (d) a heatmap given by the attention mechanism, (e) a heatmap given by the Instance+max. We rescaled the attention weights and instance scores using $a'_k = (a_k - \min(\mathbf{a}))/(\max(\mathbf{a}) - \min(\mathbf{a}))$.

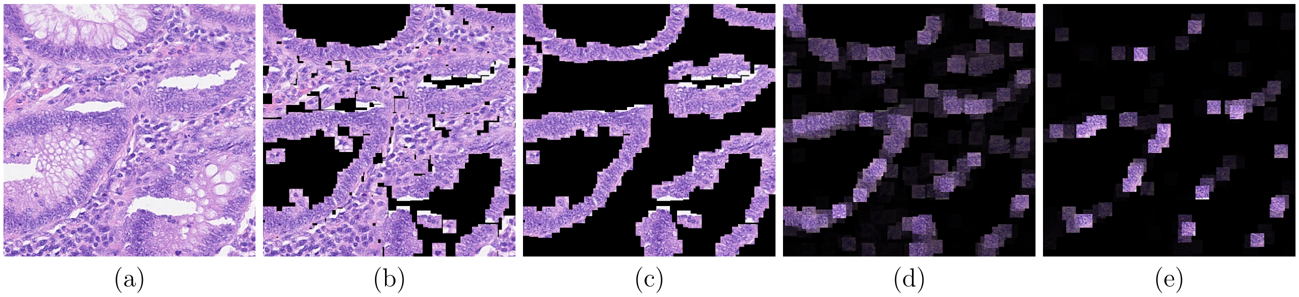

Figure 10. Colon cancer example 1: (a) H&E stained histology image. (b) 27×27 patches centered around all marked nuclei. (c) Ground truth: Patches that belong to the class epithelial. (d) Attention heatmap: Every patch from (b) multiplied by its attention weight. (e) Instance+max heatmap: Every patch from (b) multiplied by its score from the INSTANCE+max model. We rescaled the attention weights and instance scores using $a'_k = (a_k - \min(\mathbf{a}))/(\max(\mathbf{a}) - \min(\mathbf{a}))$.

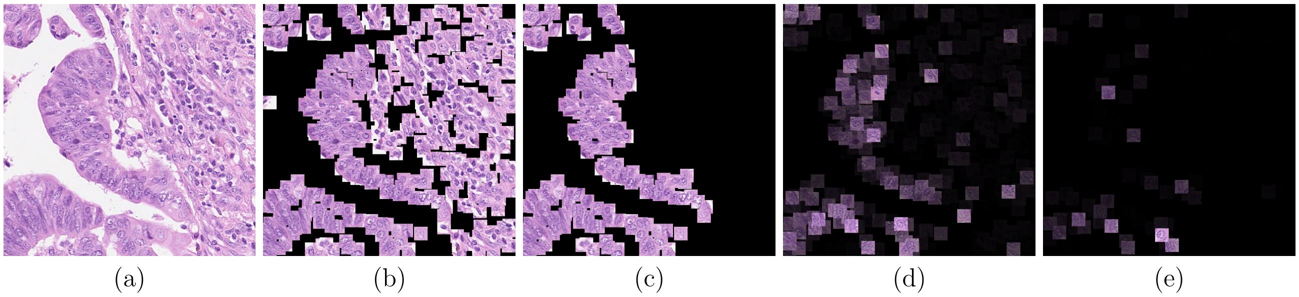

Figure 11. Colon cancer example 2: (a) H&E stained histology image. (b) 27×27 patches centered around all marked nuclei. (c) Ground truth: Patches that belong to the class epithelial. (d) Attention heatmap: Every patch from (b) multiplied by its attention weight. (e) Instance+max heatmap: Every patch from (b) multiplied by its score from the INSTANCE+$\max$ model. We rescaled the attention weights and instance scores using $a'_k = (a_k - \min(\mathbf{a}))/(\max(\mathbf{a}) - \min(\mathbf{a}))$.

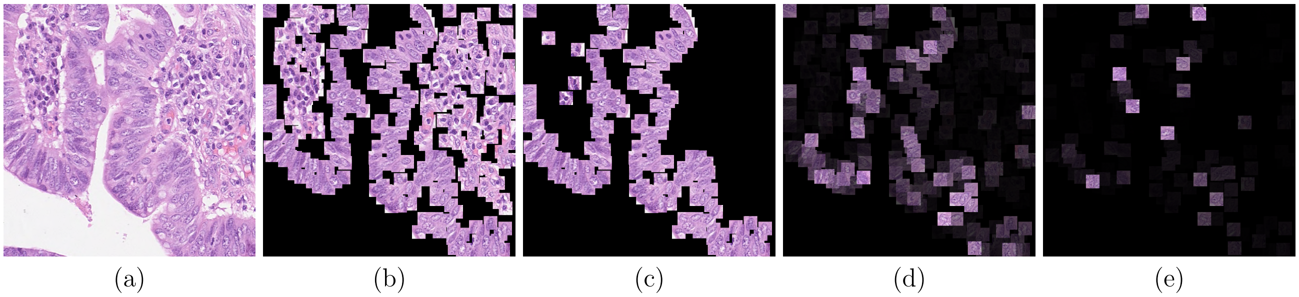

Figure 12. Colon cancer example 3: (a) H&E stained histology image. (b) 27×27 patches centered around all marked nuclei. (c) Ground truth: Patches that belong to the class epithelial. (d) Attention heatmap: Every patch from (b) multiplied by its attention weight. (e) Instance+max heatmap: Every patch from (b) multiplied by its score from the INSTANCE+$\max$ model. We rescaled the attention weights and instance scores using $a'_k = (a_k - \min(\mathbf{a}))/(\max(\mathbf{a}) - \min(\mathbf{a}))$.