Multiple Instance Learning: Attention to Instance Classification

Denis Baručić and Jan Kybic

Faculty of Electrical Engineering, Czech Technical University in Prague

ABSTRACT

Multiple instance learning (MIL) is a specific form of weakly-supervised learning where instances with hidden labels are grouped into bags, and only bag labels are observed. MIL models generally fall into one of two classes, focusing on instance or bag classification. Blurring the line between the two classes, an existing attention-based MIL method classifies bags accurately while indicating key instances. We build upon this method and propose to jointly learn a bag and instance classifier, essentially removing the distinction between bag-centric and instance-centric approaches. We performed experiments on the CAMELYON16 dataset of histopathological images and two other image datasets. The experiments showed that our method achieves high bag-level performance, comparable to other competing MIL methods. At the same time, our method outperforms other MIL methods in instance-level classification and, when provided with enough data, achieves results comparable to supervised learning using instance labels.

Keywords: Machine Learning, Multiple Instance Learning, Instance Classification

1. INTRODUCTION

Standard supervised learning assumes a training set of annotated instances. However, instance annotations are not always readily available, leading to weakly-supervised learning, in which annotations of groups of instances are provided instead. A form of weakly-supervised learning is *multiple instance learning* (MIL). In MIL, the learner is given training instances divided into groups called *bags*. The goal is to produce a bag classifier or an instance classifier (or both), having access only to binary bag labels. While hidden, the instance labels determine the bag labels: a bag is positive if and only if it contains at least one positive instance.

MIL was first introduced in the context of predicting drug activity[1,2] and has been an active research topic ever since. MIL applications range from predicting gene binding sites in RNA[3] through text sentiment analysis[4] to image classification,[5] object detection,[6] and more. A notable application is classification of whole-slide images of tissue for histopathology.[7]

1.1 MIL approaches

Assume a bag $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of n instances, where each instance is associated with a hidden label $y_i \in \{0, 1\}$. The bag is annotated with a bag label $Y \in \{0, 1\}$ such that

$$Y = \max_i y_i. \tag{1}$$

An instance is called *key* (or *witness*) if $y_i = Y$ or *non-key* if $y_i \neq Y$. We define a binary vector $\mathbf{t} \in \{0, 1\}^n$ that indicates the key instances:

$$t_i = \begin{cases} 1 & \text{if } y_i = Y, \\ 0 & \text{if } y_i \neq Y. \end{cases} \tag{2}$$

MIL models generally take one of the two approaches: *instance-level* or *embedding-level*. In both cases, the classifier can be written as

$$g\Big(\mathcal{A}\big\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)\big\}\Big), \tag{3}$$

E-mail: barucden@fel.cvut.cz (D.B.), kybic@fel.cvut.cz (J.K.)

where f and g are suitable transformations, and $\mathcal{A}$ is a permutation-invariant operator such as mean or maximum. In the instance-level approach, f is an instance classifier that produces a score for each instance, $\mathcal{A}$ aggregates the individual scores, and g is identity. In contrast, the embedding-level approach employs f to project each instance into a low-dimensional feature space. $\mathcal{A}$ then aggregates the vectors to form a bag embedding, which is finally classified by a bag classifier g. When $\mathcal{A}$ is permutation-invariant, the general form in (3) is a set function,[8,9] which is suitable for MIL as MIL assumes no ordering of instances within bags. Compared to instance-level approaches, embedding-level approaches generally promise better bag-level performance. On the other hand, instance-level approaches can be directly used to classify instances.[10]

Combining both aforementioned approaches, Ilse et al.[10] proposed a method that follows the embedding-level approach but also assigns a relevance score to each instance via an attention mechanism. Due to the employment of the attention mechanism, the method is called Attention-based MIL (AMIL).

AMIL achieves high bag-level classification performance but cannot be directly applied to instances. In this paper, we propose an extension of AMIL that maintains the same model but employs a different learning objective, focusing on both bag and instance classification. We hypothesize that bag- and instance-level classification complement each other, so we expect our changes to improve instance classification performance while maintaining the performance on bags. We show empirically that this is indeed the case.

2. METHOD

We first describe AMIL's model and learning objective[10] since our method is an extension thereof. Then, we propose a loss function that explicitly focuses on instance classification. Finally, we present the final learning criterion that combines the bag-level loss from AMIL and the proposed instance-level loss.

2.1 Attention-based MIL

Given a bag $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, AMIL encodes the instances as D-dimensional feature vectors,

$$\mathbf{h}_i = f(\mathbf{x}_i) \in \mathbb{R}^D, \quad i = 1, \ldots, n, \tag{4}$$

where f is a deep network with D outputs in the last layer. The feature vectors are then aggregated via a weighted average to obtain a bag embedding,

$$\mathbf{z} = \sum_{i=1}^{n} a_i \, \mathbf{h}_i. \tag{5}$$

The weights $a_i \in [0, 1]$ control the contribution of each instance in a bag and can, therefore, be used to identify the key instances that influence the bag label the most. They are obtained through softmax normalization,

$$a_i = \frac{\exp(r_i)}{\sum_{j=1}^{n} \exp(r_j)}, \tag{6}$$

applied on the output of a two-layer neural network*,

$$r_i = \mathbf{w}^\top \tanh(\mathbf{V}\mathbf{h}_i), \tag{7}$$

where $\mathbf{w} \in \mathbb{R}^L$ and $\mathbf{V} \in \mathbb{R}^{L \times D}$ are learnable parameters. The bag embedding $\mathbf{z}$ is passed to a classifier $g\colon \mathbb{R}^D \to \mathbb{R}$ that predicts a bag-level score, which is then transformed using the sigmoid function σ to obtain a probability that the bag is positive,

$$p = \frac{1}{1 + \exp(-g(\mathbf{z}))} = \sigma(g(\mathbf{z})). \tag{8}$$

Classifier g can be any differentiable function. We used a single linear layer, which is a common choice.[10] A bag label prediction is obtained by thresholding the probability, e.g., $p \geq 0.5$. The model is trained end-to-end by minimizing the bag-level logarithmic loss on the training data,

$$\ell_{\text{bag}}(p, Y) = -Y \log(p) - (1 - Y) \log(1 - p) \tag{9}$$

where $Y \in \{0, 1\}$ is the ground-truth bag label.

*A more complex network was also proposed, but we do not consider it here since it achieved comparable results.

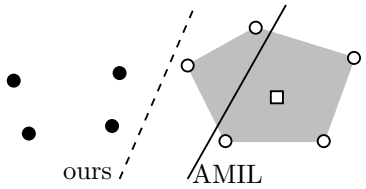


Figure 1. AMIL identifies key instances (○) in the bag by assigning them non-zero attention. On the other hand, non-key instances (●) are assigned (almost) zero attention. The bag embedding (□) thus lies somewhere in the convex hull of the key instances (gray area). AMIL then aims to set its decision boundary (solid line) to classify the bag embedding as positive. Our method shares that goal but also tries to classify the individual key instances as positive, leading to a decision boundary with an improved margin (dashed line).

2.2 Instance classification

Let us consider the task of learning an instance classifier. The bag representation $\mathbf{z}$ (5) is a convex combination of the instance feature vectors $\mathbf{h}_i$. It is, therefore, reasonable to consider employing the same classifier g to classify not only bags but also instances (see Figure 1). We denote the instance-level probability predictions of the instances being positive

$$q_i = \sigma(g(\mathbf{h}_i)), \quad i = 1, \ldots, n. \tag{10}$$

Since optimizing the bag-level loss alone does not necessarily yield good instance-level predictions (as shown later in Section 4.1), one possible strategy is to focus on instance classification during learning. If the instance labels y_i were available, the standard, supervised way of learning would be to minimize the instance-level loss,

$$\ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} \ell^{\gamma}_{\log}(q_i, y_i), \tag{11}$$

where

$$\ell^{\gamma}_{\log}(q_i, y_i) = -\frac{\gamma}{1+\gamma} y_i \log(q_i) - \frac{1}{1+\gamma}(1 - y_i) \log(1 - q_i) \tag{12}$$

is a weighted logarithmic loss function with a parameter $\gamma \geq 0$ that controls the weight of positive instances (see Section 2.2.1). The weights for positive and negative instances add up to one so that the magnitudes of the instance-level loss (11) and bag-level loss (9) are comparable.

Using the indicator vector $\mathbf{t}$ (2), we decompose (11) into losses on the key and non-key instances,

$$\ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{t}, Y) = \frac{1}{n} \sum_{i=1}^{n} t_i \, \ell^{\gamma}_{\log}(q_i, Y) + (1 - t_i) \, \ell^{\gamma}_{\log}(q_i, 1 - Y). \tag{13}$$

The indicator vector $\mathbf{t}$ is generally unknown. However, if $Y = 0$, then $t_i = 1, \forall i$, since negative bags consist purely of negative instances (1). If $Y = 1$, we propose to approximate t_i with $\pi_i(\mathbf{r})$, which is the conditional probability that the i-th instance is key that we model as follows:

$$\pi_i(\mathbf{r}) = \Pr(y_i = Y \mid \mathbf{X}) = \exp(r_i - \max_j r_j), \tag{14}$$

where r_i is defined in (7). Note that $\pi_i(\mathbf{r})$ and a_i are normalized versions of each other since $\pi_i(\mathbf{r}) = \frac{a_i}{\max_j a_j}$ and $a_i = \frac{\pi_i(\mathbf{r})}{\sum_{j=1}^{n} \pi_j(\mathbf{r})}$, $i = 1, \ldots, n$ (π_i are normalized so that the maximum is one, while a_i are normalized so that their sum is one). With the plug-in estimate in place, we obtain the instance-level loss function

$$\ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{r}, Y) = \frac{1}{n} \sum_{i=1}^{n} \ell^{\gamma}_{\log}(q_i, Y \pi_i(\mathbf{r})), \tag{15}$$

where we take the liberty to abuse the notation since $\ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{r}, Y)$ is an approximation of $\ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{t}, Y)$ from (13).

In summary, we approach learning an instance classifier from weak, bag-level labels by iterative supervised learning with instances annotated by the pseudo-labels $Y\pi_i(\mathbf{r})$. When training the network, we apply the stop-gradient operation on $\mathbf{r}$ in (15) to avoid gradient propagation through the pseudo-labels.[11]

2.2.1 Instance weights

By the MIL assumption (1), any negative bag contains only negative instances. On the other hand, positive bags often contain only a few positive instances. As a result, MIL datasets are often strongly imbalanced at the instance level, which can impact the performance when training an instance classifier. To compensate for the imbalance, we control the contribution of positive instances via the weight γ (12).

When processing a positive bag, we estimate the proportion of positive instances,

$$\beta = \frac{1}{n} \sum_{i=1}^{n} \pi_i(\mathbf{r}), \tag{16}$$

and use it to estimate the total number of positive and negative instances in the dataset,

$$n^+ = \alpha m \cdot \beta n, \tag{17a}$$

$$n^- = \alpha m \cdot (1-\beta) n + (1-\alpha) m \cdot n, \tag{17b}$$

where α denotes the proportion of positive bags, and m is the number of training bags. Finally, to balance the contribution of both classes, we set the weight γ in (12) for each bag to

$$\gamma = \begin{cases} \frac{n^-}{n^+} = \frac{1-\alpha\beta}{\alpha\beta} & \text{if } Y = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Note that γ is bag-specific.

2.3 Combining losses

The final learning objective is obtained by adding the bag- and instance-level losses together,

$$L(p, \mathbf{r}, \mathbf{q}, Y) = \ell_{\text{bag}}(p, Y) + \xi_T \, \ell^{\gamma}_{\text{inst}}(\mathbf{q}, \mathbf{r}, Y), \tag{19}$$

where factor ξ_T controls the contribution of the instance-level loss in epoch T. The accuracy of the pseudo-labels used in the instance-level loss (15) depends on the model's ability to identify key instances. Since the model learns to recognize the key instances only gradually by optimizing the bag-level loss (9), we believe that the influence of the instance-level loss on the learning objective should gradually increase in a controlled way. To this end, we propose a simple scheduling scheme,

$$\xi_T = \min\{\xi_0 \cdot q^{T-1}, \xi_{\max}\}. \tag{20}$$

The initial factor $\xi_0 > 0$ is set to a small value, and $q \in \mathbb{R}^+$ is selected empirically. We used $\xi_0 = 10^{-3}$ and $q = 2$ in our experiments. The maximum factor, $\xi_{\max}$, can be used to shift the focus between the bag and instance level. We use $\xi_{\max} = 1$ since it promises the best trade-off according to our experiments.

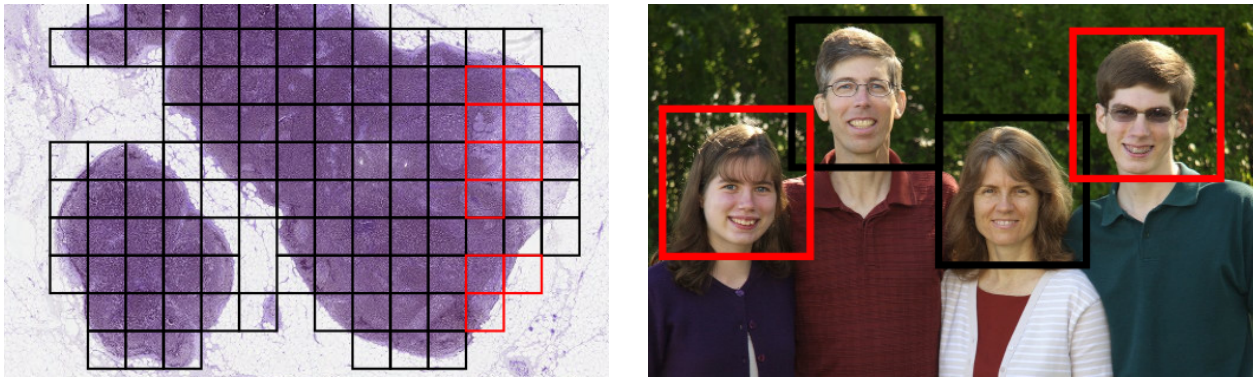
3. RELATED WORK

Early MIL approaches based on deep learning employed a non-learnable MIL pooling operator, e.g., maximum[12] or a soft approximation thereof.[13] AMIL[10] brought up a learnable pooling operator, which proved very effective and became a popular component in subsequent MIL methods.[14–19]

Our method is a direct extension of AMIL, focusing on instance classification. Instance classification has been considered in several works. Namely, mi-Net[20] classifies bags by aggregating instance predictions via a fixed MIL pooling such as maximum, mean, or LogSumExp. Like us, Zhu et al.[12] model instance-level probabilities and optimize an instance-level loss. However, they assume a fixed number of key instances per bag and consider it a hyper-parameter. Javed et al.[7] proposed an adjusted pooling scheme called Additive MIL (Add-MIL), which, in contrast to AMIL (8), predicts bags as

$$p = \sigma\Big(\sum_{i=1}^{n} g(a_i \, \mathbf{h}_i)\Big). \tag{21}$$

Several approaches consider instance-level optimization to refine the extracted features, employing, e.g., clustering[21,22] or contrastive learning.[19]



(A) CAMELYON16 (B) Faces

Figure 2. Example images from two datasets with delineated patches. The black and red patches are negative and positive instances, respectively. All patches within one image form a bag. The example (A) from CAMELYON16 is a cropped sample from a bigger image.

4. EXPERIMENTS

We experimentally compare our method with *AMIL*[10] and two other MIL methods suited for instance classification, *mi-Net*[20] and *Add-MIL*.[7] Multiple pooling operators were proposed for mi-Net; we employed LogSumExp since it performed the best according to our experiments. Apart from the MIL methods, we performed the same experiments with the standard fully-supervised method trained using instance labels. The fully-supervised method classified a bag according to (1). We considered three image datasets:

MNIST-MIL The MNIST-MIL dataset is based on the well-known dataset of hand-written digits. We create a MIL problem:[10] the MNIST images are randomly drawn with replacement to form bags of size $n \sim \mathcal{N}(100, 20)$ sampled from a Gaussian distribution (and rounded up to the nearest integer), ensuring exactly half of the generated bags contain at least one digit nine, which is selected as the positive class. Training and testing bags are guaranteed to share no instances. We employ LeNet-5,[23] a small convolutional network, as the feature encoder f.

CAMELYON16 CAMELYON16[24] is a large dataset of lymph-node-metastasis histology images of breast tissue. Out of the total of 400 images, 159 contain tumor tissue and are considered positive. The dataset provides tumor tissue annotations in the form of contour lines. We extract non-overlapping 256×256 px patches from each image at the $20\times$ magnification level, skipping those containing only the background. As a result, we obtain on average 8990 patches per image. A positive image yields about 900 positive patches. Employing a ResNet-50 pre-trained on ImageNet, the patches are encoded as 1024-dimensional vectors[22] to be used as an input of the network f. The network f consists of a single linear layer with 512 neurons.

Faces The last dataset consists of faces extracted from group photos.[25] Every photo is provided with the eye coordinates of all depicted faces, each annotated by an estimate of the person's age[†]. Using the eye coordinates, we extract all faces and label them based on the age estimates as *juvenile* (under twenty years old) or *adult*, which we consider as the positive or negative class, respectively. The faces from a single image form a bag, and each face is encoded as a vector using the same procedure as in the case of CAMELYON16. Only bags of at least five instances are kept to make the learning problem more difficult. Ultimately, we obtain 1059 positive and 1330 negative bags with a median size of seven instances. We use the same architecture of f as for CAMELYON16.

Each experiment is repeated five times, shuffling the data each time. All methods employ the same backbone architectures and use the same data in every experiment to ensure a fair comparison. All experiments were performed on a computer with an Intel Xeon Scalable Gold 6150 (18 cores, 2.7 GHz), a single NVIDIA Tesla V100 (32 GB), and 64 GB memory. Our implementation is available online[‡].

[†] http://chenlab.ece.cornell.edu/people/Andy/ImagesOfGroups.html

[‡] https://github.com/barucden/mil-atic

Table 1. Testing AU-ROC at the bag and instance level on three datasets. The values indicate the mean and its standard error computed over five runs.

	MNIST-MIL		CAMELYON16		FACES	
method	bag	instance	bag	instance	bag	instance
ours	100.00 ± 0.00	97.72 ± 0.16	87.87 ± 2.78	92.33 ± 1.22	87.36 ± 0.66	77.44 ± 0.38
AMIL	100.00 ± 0.00	96.24 ± 0.88	87.73 ± 2.31	88.84 ± 2.41	87.37 ± 0.59	76.50 ± 0.35
Add-MIL	99.99 ± 0.01	93.64 ± 1.62	80.96 ± 5.34	77.91 ± 6.75	87.05 ± 0.66	75.97 ± 0.48
mi-Net	100.00 ± 0.00	98.38 ± 0.13	79.79 ± 3.80	90.63 ± 1.28	86.82 ± 0.60	77.70 ± 0.22
supervised	100.00 ± 0.00	99.92 ± 0.01	85.72 ± 2.01	94.70 ± 0.99	86.09 ± 0.71	82.57 ± 0.30

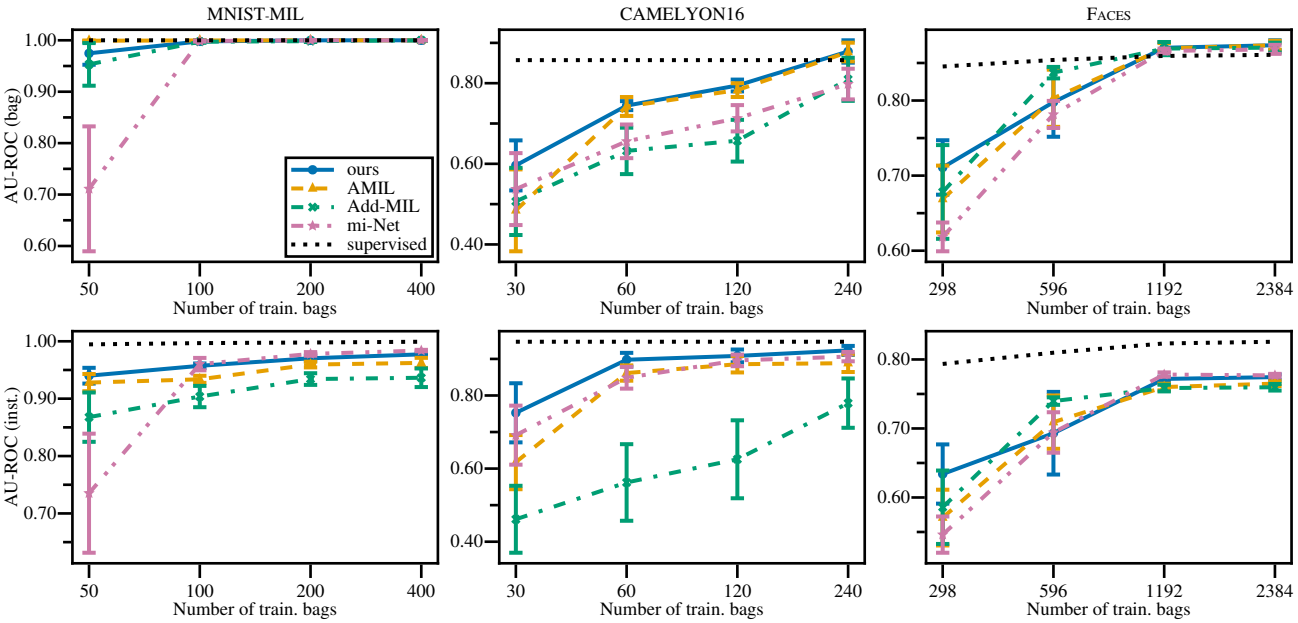


Figure 3. Testing bag-level and instance-level AU-ROC for different sizes of the training dataset. The plots show mean values and standard errors.

4.1 Results

We compare the methods in terms of the Area under the Receiver Operating Characteristic Curve (AU-ROC) computed at both bag and instance level (see Table 1). In general, our method improves the instance-level performance over AMIL while maintaining AMIL's bag-level performance, which is already relatively high. Indeed, none of the other tested methods, including the supervised method, provided a better bag classifier than our method or AMIL. We remark that in the original manuscript,[7] Add-MIL outperformed AMIL in terms of bag-level performance, which we could not replicate. In our experiments, our method consistently outperformed Add-MIL in terms of both bag and instance classification. The instance AU-ROC on CAMELYON16 is generally higher than the bag AU-ROC, which is expected since the dataset contains mostly negative instances.[26]

Although the instance-level performance of mi-Net on MNIST-MIL and Faces was slightly better, our method reached superior performance on CAMELYON16. We believe that CAMELYON16 is a challenging dataset for mi-Net due to the low rate of positive instances in the bags.

To compare their generalization ability, we trained the competing models on several subsets of the training data and evaluated them on the original testing data. Our method was consistently among the best-performing models in terms of both bag- and instance-level performance (see Fig. 3). In particular, on CAMELYON16, our method exceeded the bag-level performance of the supervised method and approached its instance-level performance when trained on 240 bags.

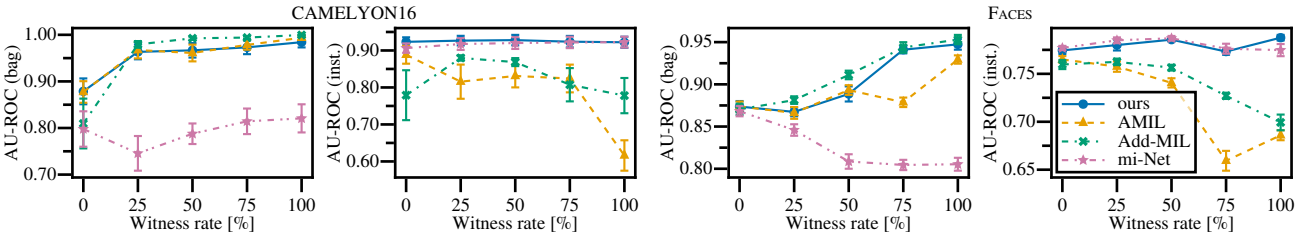


Figure 4. Testing bag-level and instance-level AU-ROC for different witness rates. The plots show mean values and standard errors.

Finally, we used the CAMELYON16 and Faces datasets to experiment with the *witness rate*, which is the proportion of key instances per bag. To ensure the requested witness rate, we altered the datasets by removing negative instances from every positive bag until the proportion of positive instances was at least the witness rate. As a result, a witness rate of zero corresponds to the original dataset, whereas one implies that all bags contain only key instances.

AMIL and Add-MIL behaved similarly: their bag-level performance improved as the witness rate increased, while the instance-level performance declined (see Fig. 4). Our method followed the bag-level performance of AMIL but maintained high performance at the instance level. Only mi-Net matched our method in terms of the instance-level classification. However, mi-Net achieved poor results at the bag level.

5. CONCLUSIONS

We proposed a deep learning method for multiple instance learning (MIL). The method builds upon Attention-based MIL (AMIL), a popular MIL method with a state-of-the-art performance that employs an attention mechanism to identify key instances.[10] Our method uses the same network architecture as AMIL but, unlike AMIL, optimizes a learning objective that also explicitly considers instance classification.

Our method significantly outperforms AMIL at the instance level, especially when bags contain a high concentration of key instances. According to our experiments, only mi-Net can match the instance-level performance of the proposed method, but it has substantially worse performance at the bag level for high witness rate. In contrast, our method delivers high performance at both the bag and the instance level. In fact, when provided with enough data, our method produces an instance classifier whose performance is comparable to the performance achieved by a classifier trained using instance labels.

The success of our method depends on the accuracy of the key instance identification, where we rely on the attention mechanism introduced by AMIL and where there is potential for improvement.

ACKNOWLEDGMENTS

The authors acknowledge the support of the OP VVV funded project "CZ.02.1.01/0.0/0.0/16_019/0000765 Research Center for Informatics", the Czech Health Research Council project NV19-08-00362, and the Grant Agency of the CTU in Prague (grant No. SGS23/118/OHK3/2T/13).

REFERENCES

[1] Dietterich, T. G., Lathrop, R. H., and Lozano-Pérez, T., "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence* **89**(1), 31–71 (1997).

[2] Maron, O. and Lozano-Pérez, T., "A framework for multiple-instance learning," in [*Advances in Neural Information Processing Systems*], **10**, 570–576 (1997).

[3] Bandyopadhyay, S., Ghosh, D., Mitra, R., and Zhao, Z., "MBSTAR: Multiple instance learning for predicting specific functional binding sites in microRNA targets," *Scientific Reports* **5**(1), 8004 (2015).

[4] Kotzias, D., Denil, M., de Freitas, N., and Smyth, P., "From group to individual labels using deep features," in [*Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*], 597–606 (2015).

[5] Hong, R., Wang, M., Gao, Y., Tao, D., Li, X., and Wu, X., "Image annotation by multiple-instance learning with discriminative feature mapping and selection," *IEEE Transactions on Cybernetics* **44**(5), 669–680 (2014).

[6] Ali, K. and Saenko, K., "Confidence-rated multiple instance boosting for object detection," in [*2014 IEEE Conference on Computer Vision and Pattern Recognition*], 2433–2440 (2014).

[7] Javed, S, A., Juyal, D., Padigela, H., Taylor-Weiner, A., Yu, L., and Prakash, A., "Additive MIL: Intrinsically interpretable multiple instance learning for pathology," in [*Advances in Neural Information Processing Systems*], **35**, 20689–20702 (2022).

[8] Qi, C. R., Su, H., Mo, K., and Guibas, L. J., "Pointnet: Deep learning on point sets for 3D classification and segmentation," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (July 2017).

[9] Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J., "Deep sets," in [*Advances in Neural Information Processing Systems*], **30** (2017).

[10] Ilse, M., Tomczak, J., and Welling, M., "Attention-based deep multiple instance learning," in [*Proceedings of the 35th International Conference on Machine Learning*], *Proceedings of Machine Learning Research* **80**, 2127–2136 (2018).

[11] Fini, E., Astolfi, P., Alahari, K., Alameda-Pineda, X., Mairal, J., Nabi, M., and Ricci, E., "Semi-supervised learning made simple with self-supervised clustering," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 3187–3197 (2023).

[12] Zhu, W., Lou, Q., Vang, Y. S., and Xie, X., "Deep multi-instance networks with sparse label assignment for whole mammogram classification," in [*Medical Image Computing and Computer Assisted Intervention – MICCAI 2017*], 603–611 (2017).

[13] Pinheiro, P. O. and Collobert, R., "From image-level to pixel-level labeling with convolutional networks," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (2015).

[14] Yao, J., Zhu, X., Jonnagaddala, J., Hawkins, N., and Huang, J., "Whole slide images based cancer survival prediction using attention guided deep multiple instance learning networks," *Medical Image Analysis* **65**, 101789 (2020).

[15] Han, Z., Wei, B., Hong, Y., Li, T., Cong, J., Zhu, X., Wei, H., and Zhang, W., "Accurate screening of COVID-19 using attention-based deep 3D multiple instance learning," *IEEE Transactions on Medical Imaging* **39**(8), 2584–2594 (2020).

[16] Li, B., Li, Y., and Eliceiri, K. W., "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*], 14318–14328 (2021).

[17] Hu, H., Ye, R., Thiyagalingam, J., Coenen, F., and Su, J., "Triple-kernel gated attention-based multiple instance learning with contrastive learning for medical image analysis," *Applied Intelligence* **53**, 20311–20326 (2023).

[18] Tang, W., Huang, S., Zhang, X., Zhou, F., Zhang, Y., and Liu, B., "Multiple instance learning framework with masked hard instance mining for whole slide image classification," in [*Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*], 4078–4087 (October 2023).

[19] Liu, K., Zhu, W., Shen, Y., Liu, S., Razavian, N., Geras, K. J., and Fernandez-Granda, C., "Multiple instance learning via iterative self-paced supervised contrastive learning," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 3355–3365 (2023).

[20] Wang, X., Yan, Y., Tang, P., Bai, X., and Liu, W., "Revisiting multiple instance neural networks," *Pattern Recognition* **74**, 15–24 (2018).

[21] Sharma, Y., Shrivastava, A., Ehsan, L., Moskaluk, C. A., Syed, S., and Brown, D., "Cluster-to-Conquer: A framework for end-to-end multi-instance learning for whole slide image classification," in [*Proceedings of the Fourth Conference on Medical Imaging with Deep Learning*], *Proceedings of Machine Learning Research* **143**, 682–698 (2021).

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

[23] LeCun, Y., Jackel, L., Bottou, L., Brunot, A., Cortes, C., Denker, J., Drucker, H., Guyon, I., Muller, U., Sackinger, E., et al., "Comparison of learning algorithms for handwritten digit recognition," in [*International Conference on Artificial Neural Networks*], **60**(1), 53–60 (1995).

[24] Bejnordi, B. E. et al., "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *JAMA* **318**(22), 2199–2210 (2017).

[25] Gallagher, A. C. and Chen, T., "Understanding images of groups of people," in [*2009 IEEE Conference on Computer Vision and Pattern Recognition*], 256–263 (2009).

[26] Davis, J. and Goadrich, M., "The relationship between Precision-Recall and ROC curves," in [*Proceedings of the 23rd International Conference on Machine Learning*], 233–240 (2006).